A COMPLETE MODEL FOR DISPLACEMENT MONITORING BASED ON UNDIFFERENCED GPS OBSERVATIONS

Johan Vium Andersson

Doctoral Thesis in Infrastructure, Geodesy
Geodesy Report No 1066

Royal Institute of Technology (KTH)
Division of Geodesy
100 44 Stockholm, Sweden
June 2008
Abstract

During recent years there has been a great focus on the climate changes within the media. More or less every day more newspaper articles are presented about the global warming issue and the effect on us human race. Climate models predict higher temperatures and more rain in the northern part of Europe. It is also predicted that the weather will become more extreme e.g. it will rain a lot during longer periods than has been the norm. If these predictions are correct, the amount of water that is going to be transported away in streams and rivers will increase and so also will the subsoil water level. The latter increases the risk for landslides in areas with fine grained soils. An early warning system that is able to alert people before a landslide take place would be of great interest.

The purpose of this work is to develop a complete real-time displacement monitoring system based on observations from several GPS-receivers that could be used as an early warning system. Due to the complex correlation structure of the traditionally used double differences, an alternative method based on undifferenced observations is used. Theoretically this approach shows some advantages and simplifies the correlative structure of observables compared to the double differenced method. A complete model for the undifferenced approach is presented in this thesis including its software implementation.

A displacement detection system includes not only the positioning algorithms, but also methods to detect if any displacement occurs. There are many methods available to discriminate displacements, which are used in the traditional control of manufacturing processes. Several of these methods are compared in this thesis, such as the Shewhart chart, different Weighted Moving Average (WMA) charts and the Cumulative SUMmation (CUSUM). Practical tests show that it is possible to detect an abrupt shift on sub centimetre level at the same epoch as the shift occurs. Smaller shifts are also detectable with the applied approach but with a slightly longer detection time.

Keywords: Undifferenced GPS observations, Real-time displacement monitoring, Early warning system, CUSUM
Acknowledgements

I wish to express my gratitude to my supervisors Professor Lars E Sjöberg and Docent Milan Horemuž, who have helped, supported and encouraged me through my entire thesis work. Their knowledge and experience have been a true help for me during the last years, and especially the last months when both of you actually pushed me forward. Without this support this thesis would not have seen the light of day.

A special thanks to Dr. Huaan Fan and Mr. Erick Assenjo, for always helping, when help was needed, and for all of the enjoyable lunch discussions we had during these years. I would also like to extend my gratitude to all of my former and present PhD colleges at the division of Geodesy for their support.

Furthermore, I wish to thank Michael Skoglund at the Swedish National Road Administration and Bo Jonsson at the National Land Survey (Lantmäteriet) for providing me with access to the ftp-archive with GPS-observations from the permanent reference stations in Gothenburg. This saved a lot of headaches and several hours of work.

I am also indebted to my colleagues at WSP Sweden AB for the encouragement to continue my studies and for letting me be on leave for all of these years, I must have established some kind of record for being on leave for almost eight years.

I want to thank the Swedish Research Council for Environment, Agricultural Sciences and Spatial Planning, FORMAS, for their financial support of this research, within the framework “Monitoring of construction and detection of movements by GPS ref no. 2002-1257”.

Also, I am grateful to the Geodetic Research Division at Lantmäteriet that in the last minute stepped in to eliminate the financial gap that occurred at the end of the project, without your support I do not think that this project would have been finished.

I want to express my gratitude to my family, father and mother, Anna, Jim, Jamie and especially you Karin, for all the support, patience and understanding and the enormous contribution that you have made to this work. A special thanks to you Jim, for proofreading the essential parts of the manuscript. Finally, I want to extend this gratitude to all of my friends for being patient with me and my absence. I promise you that I will spend more time together with you all in the future.
## Content

1  Introduction .......................................................................................................................... 1

1.1  Author’s Contributions ..................................................................................................... 6

2  Materials and Methods ......................................................................................................... 7

2.1  Adjustment procedures ..................................................................................................... 7

   2.1.1  Least square adjustment .......................................................................................... 7

   2.1.2  Sequential Least Square Adjustment ........................................................................ 8

   2.1.3  The Kalman filter ...................................................................................................... 10

   2.1.3.1  The Prediction Step ............................................................................................ 10

   2.1.3.2  Summary of the Kalman filter algorithm ............................................................. 13

   2.1.3.3  Quality control ..................................................................................................... 15

       2.1.3.3.1  Quality control in Kalman filter ................................................................. 16

2.2  Positioning with GPS ......................................................................................................... 20

   2.2.1  Observation equations ............................................................................................... 20

   2.2.1.1  Code Pseudorange observations ........................................................................ 21

   2.2.1.2  Phase observations ............................................................................................. 22

   2.2.1.3  Biases and noise ................................................................................................. 23

   2.2.2  Positioning methods .................................................................................................. 27

       2.2.2.1  Phase differences .............................................................................................. 28

           2.2.2.1.1  Single differences ...................................................................................... 28

           2.2.2.1.2  Double differences .................................................................................... 30

           2.2.2.1.3  Correlations ............................................................................................... 31

           2.2.2.1.4  Correlations in single differences ............................................................... 31

           2.2.2.1.5  Single differences in a multi-station solution ................................................. 32

           2.2.2.1.6  Correlations in double differences .............................................................. 33

           2.2.2.1.7  Double differences in a multi-station solution ............................................. 34

           2.2.2.1.8  Correlation in time ...................................................................................... 35
2.2.2.2 Undifferenced solution………………………………………………………………………35
2.2.2.3 Double Differences vs. Undifferenced data…………………………………………………36

2.3 State vector models………………………………………………………………………………38
2.3.1 Position and velocity……………………………………………………………………………39
2.3.2 Receiver clock……………………………………………………………………………………41
2.3.3 Atmospheric delays……………………………………………………………………………43
2.3.3.1 Ionosphere……………………………………………………………………………………43
2.3.3.1.1 Deterministic delay…………………………………………………………………………44
2.3.3.1.2 Parameter modelling………………………………………………………………………46
2.3.3.2 Troposphere……………………………………………………………………………………47
2.3.3.2.1 Deterministic model………………………………………………………………………49
2.3.3.2.2 Mapping functions………………………………………………………………………50
2.3.3.2.3 Parameter modelling……………………………………………………………………51
2.3.4 Receiver antenna and Multipath………………………………………………………………52
2.3.4.1 Receiver antenna………………………………………………………………………………52
2.3.4.2 Multipath……………………………………………………………………………………55
2.3.5 Common errors…………………………………………………………………………………56
2.3.6 Observation equations…………………………………………………………………………57
2.3.6.1 At the reference stations……………………………………………………………………57
2.3.6.2 At the rover stations…………………………………………………………………………60

2.4 Software implementation………………………………………………………………………61
2.4.1 Initialisation……………………………………………………………………………………65
2.4.2 Pre-processing…………………………………………………………………………………65
2.4.2.1 Satellite positions………………………………………………………………………………66
2.4.2.2 Observation weighting………………………………………………………………………69
2.4.2.2.1 Weighting methods………………………………………………………………………70
2.4.2.3 Cycle slip detection……………………………………………………………………………71
2.4.2.3.1 Single-frequency phase / code combinations……………………………………………73
Experiments and Results .............................................................. 73

2.4.2.3.2 Dual-frequency phase combinations ................................ 73
2.4.2.3.3 Geometry free solution .................................................... 73
2.4.2.3.4 Predicted residuals .......................................................... 74

2.4.3 Fill in matrices L, H and R .................................................... 76
2.4.4 Quality control ................................................................. 79
2.4.5 Ambiguity fixing ............................................................... 81
2.4.6 Output parameters ............................................................. 82

2.5 Automatic Displacement Monitoring Algorithms ...................... 82
2.5.1 Design issues ................................................................. 84
2.5.2 Autocorrelation ............................................................... 86
2.5.2.1 Methods to remove autocorrelation .................................. 87
2.5.2.1.1 Increase the observation lag ........................................ 88
2.5.2.1.2 Model the correlative structure .................................... 88

2.5.3 Shewhart charts .............................................................. 91
2.5.4 Weighted Moving Average charts ....................................... 93
2.5.5 Exponentially Weighted Moving average ................................ 95
2.5.6 CUSUM charts ............................................................... 97
2.5.6.1 Description of CUSUM charts .......................................... 97
2.5.6.2 Influence of deviations in the statistical properties on the performance .......... 100
2.5.6.3 Optimising a CUSUM performance .................................... 101
2.5.6.3.1 Fast initial response .................................................. 101
2.5.6.3.2 Self starting CUSUM .................................................. 101

2.5.7 Comparing the charts ....................................................... 103

3 Experiments and Results ............................................................ 105

3.1 Observations ........................................................................ 105

3.2 Positioning performance test ................................................. 107
3.2.1 Single frequency observations ........................................... 108
3.2.2 Dual-frequency observations ............................................. 117
3.3  Displacement detection.................................................................121

3.3.1  Tests with simulated observations..............................................121

3.3.1.1  Test with Shewhart chart......................................................122

3.3.1.2  Weighted Moving Average charts .......................................124

3.3.1.3  EWMA ..................................................................................125

3.3.1.4  The CUSUM chart ...............................................................127

3.3.2  Tests with real observations......................................................128

3.3.2.1  Decorrelation of observations ..............................................131

3.3.2.2  Testing CUSUM charts in displacement monitoring..............134

4  Summary and Conclusions...............................................................146

4.1  Further research............................................................................151

5  References.......................................................................................153
List of tables

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Table 1</td>
<td>Additional biases that affect the GPS observables</td>
<td>24</td>
</tr>
<tr>
<td>Table 2</td>
<td>Summary of the efficiency of different deterministic ionospherical models, Klobuchar (2001)</td>
<td>44</td>
</tr>
<tr>
<td>Table 3</td>
<td>Template for the NGS calibration parameters</td>
<td>54</td>
</tr>
<tr>
<td>Table 4</td>
<td>Initial values used in the Kalman filter</td>
<td>66</td>
</tr>
<tr>
<td>Table 5</td>
<td>Comparison: Diffraction and Multipath effects (Wanninger et al. 1999)</td>
<td>69</td>
</tr>
<tr>
<td>Table 6</td>
<td>Different weighting models in the kinematic calculations of baselines. Model 1 is equal weighting, model 2 elevation dependent weighting and model 3 is C/No based weighting</td>
<td>71</td>
</tr>
<tr>
<td>Table 7</td>
<td>Influence of different values of the autocorrelation coefficient $\rho$ on the ARL for a process with ARL 1000. (Table from Hawkins and Olwell 1998 pp.77)</td>
<td>101</td>
</tr>
<tr>
<td>Table 8</td>
<td>Estimated coordinates in static mode (WGS84)</td>
<td>108</td>
</tr>
<tr>
<td>Table 9</td>
<td>The coordinate differences in the calculated mean values of kinematic and static solutions are presented together with the standard deviations of the kinematic positions</td>
<td>110</td>
</tr>
<tr>
<td>Table 10</td>
<td>Estimated coordinates in static mode (P-model) together with their corresponding standard deviations, given in WGS84</td>
<td>111</td>
</tr>
<tr>
<td>Table 11</td>
<td>Coordinate differences in the calculated mean values in kinematic and static solution is presented together with the standard deviation of the kinematic positions</td>
<td>111</td>
</tr>
<tr>
<td>Table 12</td>
<td>This table shows the results from positioning with single and dual-frequency observations in the UGPS-software</td>
<td>120</td>
</tr>
<tr>
<td>Table 13</td>
<td>Estimated autocorrelation in decorrelated observations</td>
<td>134</td>
</tr>
<tr>
<td>Table 14</td>
<td>Out-of-control ARL values for different shift sizes using in-control ARL of 900 epochs</td>
<td>135</td>
</tr>
<tr>
<td>Table 15</td>
<td>Out-of-control ARL values for different shift sizes using in-control ARL of 1800 epochs</td>
<td>135</td>
</tr>
<tr>
<td>Table 16</td>
<td>Out-of-control ARL values for different shift sizes using in-control ARL of 3600 epochs</td>
<td>135</td>
</tr>
<tr>
<td>Table 17</td>
<td>Average Run Length values for a shift of $0.1667\sigma$ and an in-control run length of 900 epochs</td>
<td>142</td>
</tr>
</tbody>
</table>
List of figures

Figure 1. Kalman filter loop (Jekeli 2001) 14
Figure 2. Single difference generated by combining observations from two receivers \(( k \text{ and } m)\), towards one satellite \(p\) 29
Figure 3. Double differences where two single differences are combined 30
Figure 4. Three receivers’ \(( k, l \text{ and } m)\) record observations within the same epoch \(t\) to one common satellite \(p\) 33
Figure 5. The parameters in the observation equations are normally changing smoothly during time like the dashed line in the figure and do not make any sudden jumps 36
Figure 6. Estimated clock drift in three receivers during 1 hour 42
Figure 7. The phase centre of an antenna does not always coincide with the physical centre. NGS determines the offsets to the antenna reference point ARP. 54
Figure 8. Observation residuals during three consecutive days, each of which is shifted in time so they represent the same epoch. Residuals of day 261 and 262 are shifted +0.005 m and +0.010 m respectively to make the plot readable 56
Figure 9. Overview of the processing scheme in the UGPS-software 64
Figure 10. General flowchart for calculating satellite positions 67
Figure 11. Fit and validity intervals 68
Figure 12. Runge’s phenomenon on a 3 hour fit interval 68
Figure 13. Influence of an undetected cycle slip 72
Figure 14. Predicted observation residuals at receiver B 75
Figure 15. By subtracting the predicted residuals for observations to two satellites it is possible to find the cycle slip 76
Figure 16. Design matrix layout for a dual-frequency situation. At the reference receiver A observations are made towards 8 satellites and at the rover 7 satellites. 80
Figure 17. A vertical displacement occurs in epoch \(k\), but it is unknown in time the epoch \(k\) when the vertical displacement occur is unknown 83
Figure 18. Control chart running in-control up until epoch 7 where it goes out of control 85
Figure 19. decorrelated observations with a 5cm shift in epoch 3000 90
Figure 20. Shift in the mean value \(\varepsilon(t) = 0.010 \text{ m}\) is introduced in epoch 1. The figure show how different correlation coefficients \((0.1, 0.5 \text{ and } 0.9)\) influences the estimated values of the process \(X(t)\) 90

x
Figure 21. the problem of inertia, CL is the centre line, UCL upper control limit and LCL lower control limit.

Figure 22. Working principles of a CUSUM chart

Figure 23. The figure shows used receiver configuration, where triangles symbolise the reference stations and the circle the area of roving receivers

Figure 24. Left picture shows the antenna of the red receiver, which is placed on top of a translation stage, and the right picture shows the green receiver placed on top of a sliding platform

Figure 25. Kinematically estimated X-coordinates (WGS84) from TTC and UGPS

Figure 26. Estimated X-coordinates in with one and two reference receivers, the solid line represents the solution with two reference receivers

Figure 27. The calculated overall statistics for the predicted residuals is growing

Figure 28. Estimated residuals in the update sequence of the Kalman filter

Figure 29. Standard deviations of the estimated coordinates, the ambiguities are fixed in epoch 200

Figure 30. The standard deviation of the estimated clock parameters and in the common errors parameters shows a typical example of estimability problems

Figure 31. Negative influence of the ionosphere is calculated with the Trimble R7 receivers but not in the data from the JAVAD receivers

Figure 32. Calculated residuals on C1 observations when using two different receiver types

Figure 33. Residuals on C1 observations when using the same receiver type

Figure 34. Simulated observations with a shift of 1 \( \sigma \) in epoch 500, the dashed line forms the mean value of the first 500 observations

Figure 35. Two Shewhart charts used on the same dataset, the upper with sample size 6 and the lower with a sample size of 36 observations

Figure 36. Comparison of two linearly WMA-charts with lag sizes 6 and 36

Figure 37. Comparing the displacement detection performance of equally and linearly weighted observations in a WMA-chart with a lag of 36 epochs

Figure 38. The influence of different values of \( \lambda \) on the performance of a EWMA chart

Figure 39. Combined EWMA charts of process mean and standard deviation for a +2\( \sigma \) shift.

Figure 40. CUSUM chart applied to the simulated observations \( k = 0.5, h = 5.071, ARL = 1000 \)
Figure 41. Principle figure of the transformation from XYZ coordinates in WGS84 to a local NEU-coordinate system

Figure 42. The displacement is not always aligned with the local coordinate axis

Figure 43. Estimated kinematic N-coordinate of station Green, without simulated displacements

Figure 44. Scatter plot with the estimated N-coordinate

Figure 45. Autocorrelation factor for different time lags

Figure 46. Scatter plot for observations with 217 epoch interval.

Figure 47. Autocorrelation factors calculated with 217 epoch interval data

Figure 48. Decorrelated N-coordinate

Figure 49. Scatter plot of the decorrelated N-coordinates

Figure 50. Real observations with a simulated step-change deformation in epoch 3000

Figure 51. Coordinates estimated with real observations, several in-field generated step-change deformations given in a local NEU coordinate system.

Figure 52. Coordinates estimated with real observations, given in a rotated local NEU coordinate system

Figure 53. Decorrelated real observations with a simulated displacement in epoch 3000

Figure 54. Outcome from a CUSUM chart, a shift is detected in epoch 3000 and thereafter is the CUSUM chart running suboptimally

Figure 55. The upper diagram shows the outcome from a CUSUM chart, a shift is detected at epoch 3000, and then the mean in Zt is updated. The original observations are plotted in the lower figure with dashed grey line and the updated mean values with the solid line

Figure 56. CUSUM chart designed to detect 5 mm shifts

Figure 57. Decorrelated observations with in-field generated step-change displacements; the upper plot shows the CUSUM chart, and the lower plot the decorrelated observations
1 Introduction

The discussion about climate change and global heating has experienced a renaissance during the last few years. More or less every day there is some ongoing discussion about the climate in media, and the issue has been brought up at the highest political level all over the world. Global and regional models have been used to study the impact of the global heating. Persson et al. (1997) used such a model to study the impact of global heating in Europe and particularly in Sweden. They concluded that the general trend is that the climate will become gradually warmer in the northern parts of Europe during winter and in the south during summer. They also predicted that the amount of rain will increase in the northern parts while the southern areas become will drier. This implies for Sweden that we can expect higher temperatures making the winter periods shorter and the precipitation during these periods will be as rain instead of snow. Furthermore, they predicted that extreme weather conditions are to be expected, e.g., stronger wind conditions, and heavy showers of rain.

Increased amounts of rain will results in a higher flooding risk. When a lot of rain arrives quickly and when it falls during longer periods, subsoil water levels will raise, which increases the risks for landslides in areas with fine-grained soils such as clay. Larger water amounts in rills and rivers will put more loads on the existing roads, railroads, hydropower dams etc., and will influence the way of designing them in the future. In other words, the predicted climate change has to be considered in urban planning to prevent loss of lives and damage to buildings or other parts of the urban infrastructure. In areas with large risk of landslides, an early warning system that sends out an alarm before a landslide occurs would be of great benefit to ensure the safety of citizens. This type of early warning system is exactly what this thesis is about.

In principle, an early warning system consists of a set of sensors that measure the displacement on the ground, a central computer that stores and analyses the data and finally, an alarm system. Local deformation processes, such as landslides or structural deformations, are very often sporadic in nature. Consequently, a high temporal or even continuous monitoring of the deformation process is necessary (Hartinger 2001). The basic assumption in these systems is that small displacements take place before a large one is triggered. A typical requirement that has to be fulfilled by an early warning system is the real-time capability. The system should be able to send out correct warnings with as short a time delay as possible.

Over the years, many different methods have been developed to perform displacement monitoring. These are based on different types of sensors and therefore classified here into four different groups: remote sensing, photogrammetric, geodetic and geotechnical techniques. These
methods can further be divided into two groups: long time episodic and real-time applications. Long time episodic measurements are displacement monitoring that is not performed in real-time but in campaigns with a given interval. On the contrary, the sensors that can be used in real-time applications are of main concern in this thesis.

Satellite remote sensing techniques have a significant potential for landslide hazard assessment and for improved understanding of landslide processes. Differential Interferometric Synthetic Aperture Radar (DINSAR) can detect deformations with sub-centimetre accuracy (Hartinger 2001). The general problem with this method is the temporal resolution limited by the orbital passages of the satellites over the actual area. This implies that this method is an episodic method and impossible to implement as a real-time method, but on the other hand it provides a good spatial coverage.

Airborne photogrammetric methods have the same problems of being episodic, but it will still be an effective method to determine long time slow moving landslides by comparing pictures taken at different epochs in time.

The geodetic methods can be divided into two subgroups; ground-based and satellite-based systems. The ground-based systems make use of levelling instruments, total-stations and laser scanners. They are usually employed according to an episodic monitoring programme where discrete points are monitored. This approach is very labour intensive, which makes it expensive to use, especially when the temporal resolution is high. To reduce the labour costs in these situations, Robotic Total Stations (RTS) are used. The RTS is programmed to automatically detect predefined targets with Automatic Target Recognition (ATR) and then perform the measurements towards them. The measured observations are then sent to a central computer that automatically performs all the displacement analyses. Typical software systems that are available on the market to perform this type of monitoring are GeoMoS developed at Leica AG Switzerland (www.leica.com), GOCA developed at the University of Applied Sciences in Karlsruhe Germany (www.goca.info) and ALERT developed at the Canadian Centre for Geodetic Engineering, see Wilkins et al. (2003). Laser scanners can be used in analogy to the total stations, but instead of measuring discrete points the shape of a surface is measured. A drawback with the ground-based geodetic methods is that it can be troublesome to perform absolute displacement analyses in situations when the distance to the fixed points are long, which they often are in high risk landslide areas. In these cases only relative monitoring can be performed, where the displacements within the investigated object is determined.

The second group of geodetic methods are the satellite-based systems. This sub-group makes use of the available Global Navigation Satellite Systems GNSS as the American Global Positioning System GPS, the Russian GLObalnaya NAvigatsionnaya Sputnikovaya Sistema GLONASS and in
the future the European GALILEO satellite positioning system. For example the GNSS-
technology has some good properties; one is that it is possible to measure baselines with a high
accuracy over long distances without any demand for line-of-sight between the receivers, which
makes absolute displacement monitoring possible.

Displacement monitoring with GNSS can be performed episodic or in real-time. The geodesy
department at the Royal Institute of Technology in Stockholm Sweden have performed episodic
measurements to determine Crustal movements in Skåne (Pan et al.2001). The accuracy in the
estimated coordinates is at \( \text{mm} \) level. Since this approach is episodic, it should be used only in
situations when there is no risks for sudden large displacements.

The last group of sensors that are used in displacement monitoring are the geotechnical sensors,
that are permanently placed on the monitored object. This group contains sensors like
extensometers used to measure changes in distances between two points, inclinometers used to
measure the slope of initially straight boreholes, piezometers used to measure the pore water
pressures and tilt meters to measure the deviation from a horizontal plane.

As can be seen, there are many different types of sensors that are used for displacement
monitoring. In this work we focus on sensors that can be used for absolute displacement
monitoring in real-time and it leaves us with a GNSS. The episodic methods should not be
rejected, as they are still very useful for displacement detection, but perhaps more suited for
long term monitoring of slow displacements.

One goal of this project is to develop a complete concept for deformation monitoring in real-time
based on GNSS observations. To limit the size of the project, only GPS observations are used and
the shifts that we try to detect are abrupt episodic shifts in the mean of the estimated
coordinates. The concept for displacement monitoring includes algorithms for both positioning
and displacement detection.

Today there are already several different displacement monitoring systems available on the
market, some of which are listed here:

- CODMS, Continuously Operating Deformation Monitoring System, developed at Graz
  University of Technology, Gassner et al. (2002)
- GOCA, (GPS based online control and alarm system) developed at Fachhochschule
- RT-MODS2- Real-Time Monitoring Of Dynamic Systems, developed at Istambul Technical
  University, Ince and Sahin (2000)
- GNPOM, Geodetic Navstar - Permanent Object Monitoring, Geo++®, (www.geopp.de)
- Motion Tracker, by Trimble (www.trimble.com)
These systems can be separated into three different categories according to their way of using the observations. The first is the post processing category where all observations are stored in files, which thereafter are calculated in batch calculations. The outcome from these batch calculations, normally a set of coordinates, are then used in some software package that performs displacement analysis. Typical software programs based on this approach are GeoMoS and Trimble Motion Tracker. These systems are not running in true real-time, but never the less their results are is useful for monitoring displacement.

The second category uses the result calculated in the receivers in RTK-mode (Real Time Kinematic) for deformation monitoring. RT-MODS2 and GOCA are typical software packages of this category. In RT-MODS2 the coordinates from the RTK solution are directly used in the deformation monitoring. GOCA uses also information that is calculated in the instruments, but instead of using the estimated coordinates, it uses the baselines. If several reference receivers are used simultaneously GOCA performs a traditional network adjustment, where the coordinates are estimated. The benefit of this approach is that outliers can be detected as well as movements in the static receivers. However, problem with this approach is that the correlations among the baselines are not treated correctly. This is a subject that is discussed in further details in Section 2.2.

The third category is the one that uses raw data; all observations are sent into the central computer, where the calculations are performed. CODMS and GNPOM are software packages that follow this approach. The differences between these software packages are that CODMS is based on double differenced observations, while GNPOM is based on undifferenced observations. Further explanation about the mathematical principles of the differenced and the undiffferenced approaches follows in the next chapter of the thesis.

All of the before mentioned software packages that are introduced above use double differenced observations, except for GNPOM which uses undifferenced observations. The main difference between the undifferenced and double differenced approach is that the double differenced method uses several observations to eliminate all common biases in the observations, belonging to the satellites and receivers, while the undifferenced approach instead of eliminating the parameters uses their correlation in time to estimate them. The key question is which of these approaches is the most attractive for displacement monitoring?: Is it the undifferenced or the double differenced approach?

The second part of a displacement monitoring system is the actual shift detection algorithm, which is an active part that sends out alarms when a shift is judged to have occurred. There are
some important properties that have to be fulfilled by this shift detection algorithm. The first is that if an alarm goes off, it should do so as fast as possible after the shift occurs. Secondly, the number of false alarms should be as low as possible, and, finally, but no less importantly, the minimal detectable shift size should be as small as possible. Several different methods to detect abrupt shifts are developed within the manufacturing industry over the years to control the quality of manufactured products, like the Shewhart-chart, different Weighted Moving Average (WMA) and the Cumulative Summation (CUSUM)-chart. These charts have already been used in monitoring applications. Ogaja (2002) used CUSUMs to detect deformations in buildings, and more recently Mertikas and Damianidis (2007) studied using CUSUMs for the detection of movement in permanent reference stations. Here the charts are applied on the estimated coordinates from the real time system. The question that arises is whether these methods are suitable for real-time displacement monitoring or not. And if so, what are their limitations?

This thesis is divided into three parts excluding this introduction; the first contains the theoretical background of the used models and their implementation whilst, the second part includes the results from practical experiments, and the final chapter contains the summary and conclusions.

The theoretical parts of this thesis are found in Chapter 2. This chapter starts with a general derivation of the Kalman filter, which constitutes the mathematical foundation in real-time kinematic positioning with GPS. This derivation is followed by a section where the positioning algorithms for double differences are compared with the undifferenced approach from a theoretical point of view. To be able to compare the methods in practice and to obtain a deeper insight to the problem that concerns the undifferenced method, a software package of the undifferential approach was developed. The used observational models and the software implementation are presented in two separate sections (2.3 and 2.4). In the final section of Chapter 2 the used displacement monitoring algorithms are introduced and compared theoretically.

The third chapter describes the outcome from practical tests with both the double differenced and undifferenced approaches. These tests are performed with both simulated and real observations. The real observations are measured in Gothenburg, and the measuring procedures are described in the first Subsection 3.1. In Section 3.2 the positioning performance of the undifferenced approach is compared with the double differenced approach in both static and kinematic calculations. Finally in Section 3.3, the quality control charts are tested to evaluate their shift detection performance.
1.1 Author’s Contributions

This thesis contains a complete description of a real-time early warning system based on GPS observations. A system like this consists of two parts: one positing part, which estimates the receiver coordinates and the other part of the system monitors the estimated positions and sends out an alarm when a displacement takes place. In this section the author’s contribution is highlighted.

The theoretical part starts with a mathematical introduction of the Kalman filter including quality control methods. Thereafter follows a comparison of the traditional double differenced method and the undifferenced method, made by the author. The outcome from these studies results in a decision to continue with the undifferenced approach in the further research.

A complete model of all unknown parameters and the software implementation is presented in Sections 2.3 and 2.4. The model and the Matlab© version of the software package is developed in collaboration with my supervisor Milan Horemuž. The new updated C++ version of the software package is developed further by the author with new algorithms for quality control. A further collaboration with Milan Horemuž was made in the development of an approach to interpolate the satellite coordinates, (Horemuž and Andersson 2006). This satellite coordinate calculation approach generalises the algorithms to calculate satellite coordinates and satellite clock corrections from broadcasted and precise ephemerides. For debugging purposes the author has developed an observation simulator that calculates the GPS observations based on given receiver positions and satellite orbits (Andersson 2006). The performance of the undiffernced approach is compared with results from a software package that uses double differenced observations. All these comparisons are also made by the author.

The second part of a displacement monitoring system consists of displacement detection algorithms. A well performing displacement detection algorithm finds a shift with a short time delay and with as few false alarms as possible. Some different models that normally used for quality control of manufacturing process are compared. Both the theoretical and the practical comparisons are performed by the author. The used methods have been used earlier by other research groups, e.g., Ogaja (2002) and Mertikas and Damianidis (2007), and they mention that the quality control algorithms are designed to use decorrelated observations, but instead of decorrelating the observations they simply ignores the autocorrelation. The autocorrelation between the estimated coordinate between two epochs in time is high and to be able to use the used quality control algorithms it is necessary to decorrelate the observations. The author shows theoretically and practically that it is possible to detect abrupt sub centimetre shifts in real-time with the introduced quality control algorithms.
2 Materials and Methods

The aim of the five subsections in this chapter is to give an overview of the material and methods that are used in this thesis. The first subsection, Section 2.1, gives an overview of the Kalman filter derivation, with a start from a traditional Least Squares Adjustment. In the following Section 2.2, the observation equations of code and phase observations are derived and the positioning algorithms for double differences are compared with the corresponding algorithms for the Undifferenced Approach.

In Section 2.3 all estimated parameters in the tested undifferenced model are described in detail and the software implementation of this model can be followed in Section 2.4. In the final subsections of this chapter some different methods for automatic shifts detection are presented. They are compared with the aim of finding the most suitable approach for automatic displacement detections.

2.1 Adjustment procedures

2.1.1 Least square adjustment

To give a proper derivation of the Kalman filter one has to start with traditional Least Squares (LSQ) adjustment that have been used in geodesy for ages. Several authors have derived the LSQ-adjustment method and described how to use it in adjustment of geodetic networks over the years, Bjerhammar (1973), Sjöberg (1984), and Koch (1999) to mention just some of them. This is why we only present the essential equations here, following the notation school of Bjerhammar.

To show the basic principle of LSQ, let us start with the following observation equation:

$$AX = I - \varepsilon$$  \hspace{1cm} (2.1)

where the design matrix $A$ is a full rank matrix that relates the unknown parameters in the vector $X$ to the observation vector $I$ and the residual vector of the observations $\varepsilon$. The observation residuals are assumed to be normally distributed with zero expectation:

$$E(\varepsilon) = 0$$ \hspace{1cm} (2.2)

and the expected variance covariance matrix is determined as:

$$E(\varepsilon\varepsilon^T) = \sigma_0^2 P^{-1} = \sigma_0^2 Q$$ \hspace{1cm} (2.3)

where the a priori unit weight standard error $\sigma_0^2$ multiplied with the inversed weight matrix $P$ or the cofactor matrix $Q$. The relation, $Q = P^{-1}$, between the cofactor matrix and weight matrix then is obvious. Observations are often assumed to be independent from each other, since it is
difficult to model the correlation between them. This assumption makes both the weight and cofactor matrix diagonal in their structure.

The relationship between the observations and unknown parameters could be both linear and non-linear. In the latter case it is necessary to linearize the observation equations before they can be adjusted. This is normally carried out using a first order Taylor series approximation. The LSQ solution to the system in Eq. (2.1) is found by minimising the estimated residuals \( \hat{e} \):

\[
\hat{e}^T P \hat{e} = \text{minimum} \tag{2.4}
\]

By doing so the final LSQ-estimates of the unknown parameters \( \hat{X} \) is given by:

\[
\hat{X} = (A^T P A)^{-1} A^T P l \tag{2.5}
\]

where the symbol “\(^\wedge\)” is used to indicate an estimated parameter. The corresponding residual vector is calculated as:

\[
\hat{e} = l - A\hat{X} \tag{2.6}
\]

and the estimated a posteriori variance factor is given by:

\[
\hat{\sigma}^2_0 = \frac{\hat{e}^T P \hat{e}}{n - m} \tag{2.7}
\]

where \( n \) and \( m \) in are the number of observations and unknown parameters, respectively. The variance-covariance matrix \( C_{\hat{X}\hat{X}} \) of the estimated unknowns and residuals \( C_{\hat{e}\hat{e}} \) can be determined by the following relations:

\[
C_{\hat{X}\hat{X}} = \hat{\sigma}^2_0 (A^T P A)^{-1} \tag{2.8}
\]

and

\[
C_{\hat{e}\hat{e}} = \hat{\sigma}^2_0 (P^{-1} - A(A^T P A)^{-1}A^T) \tag{2.9}
\]

### 2.1.2 Sequential Least Square Adjustment

LSQ-adjustment can be made directly as given in Eq.(2.5), but sometimes it is convenient to split the observations into \( k \) independent sub groups and stepwise estimate the unknowns. This procedure is known as a sequential adjustment and will give exactly the same result as the direct approach. The sequential adjustment simplifies the adjustment procedure in some situations. Typical examples are when adjusting a large geodetic network or when an existing network is expanded. Instead of readjusting the complete network when new observations are added, it is possible to use the result from the previous adjustments as pseudo observations in the new adjustment. However, this is not the only property that makes this approach attractive. The computation burden to solve large equation systems with LSQ-adjustment is very high. This is
mainly caused by the matrix inversion algorithms. By using the sequential adjustment approach when the matrices are large, it is possible to reduce the computational burden and thereby increase the computational speed, Koch (1999).

Assuming that the subgroups are independent gives the following properties to the observation weights:

\[ E\{e_k e_i^T\} = \begin{cases} P_k, & i = k \\ 0, & i \neq k \end{cases} \quad (2.10) \]

where \( k \) and \( i \) represent different subgroup numbers.

The sequential adjustment procedure starts with an initial adjustment using the first group of observations following equations:

\[ \hat{X}_{(2)} = (A_{(1)}^T P_{(1)} A_{(1)})^{-1} A_{(1)}^T P_{(1)} I_{(1)} \quad (2.11) \]

and

\[ C_{\hat{X}\hat{X}^{(1)}} = \sigma^2_0 (A_{(1)}^T P_{(1)} A_{(1)})^{-1} = \sigma^2_0 Q_{\hat{X}^{(1)}} \quad (2.12) \]

where \( Q_{\hat{X}^{(1)}} \) is regarded as the cofactor matrix of \( \hat{X}^{(1)} \). The subscript (1) indicates the sequential number of the adjustment. The recursive estimation of unknown parameters in step \( k \) is performed as:

\[ \hat{X}_k = \hat{X}_{k-1} + K_k (I_k - A_k \hat{X}_{k-1}) \quad ; k = 1,2, ... \quad (2.13) \]

where the matrix \( K_k \) is called the gain or blending matrix, that optimally distributes the weights between the pseudo observations and new observations. It is given by:

\[ K_k = (Q_{X,k-1})^{-1} A_k^T \left[ P_k^{-1} + A_k^T (Q_{X,k-1})^{-1} A_k \right]^{-1} \quad (2.14) \]

that optimally, in least square sense, blends the pseudo observations with the actual observations. The cofactor matrix \( Q_{X,k} \) in the current adjustment step is determined as:

\[ Q_{X,k} = A_k^T P_k A_k + Q_{X,k-1} \quad (2.15) \]

and the corresponding variance covariance matrix:

\[ C^{(k)}_{\hat{X}\hat{X}} = \sigma^2_0 (Q_{X,k})^{-1} \quad (2.16) \]

The recursive pattern of the sequential approach makes it suitable for real-time applications where observations are collected epoch by epoch. By using traditional LSQ adjustment every time a new set of observations is collected, it is likely that performance problems will occur caused by the computational load. A further development of the sequential adjustment for real-time applications is the Kalman filter, that will be discussed in detail in the following section.
Complete derivations of the sequential adjustment can be found in Bjerhammar (1973) and Koch (1999).

### 2.1.3 The Kalman filter

In real-time systems, the observations are collected sequentially and adjustment procedures as the sequential adjustment can be used to estimate values of some unknown parameters, epoch by epoch. This approach is very useful since it keeps the size of the matrices and the computational load down, but it should be borne in mind, that it assumes that the unknown parameters do not change in time. This property makes the sequential adjustment approach less attractive when the actual unknown parameters are part of a dynamic system.

To overcome this problem, Kalman (1960) introduced a modified algorithm of the sequential adjustment. He uses additional models that describe the temporal dynamic behaviour of each unknown parameter. These models are then used to predict the unknown parameters from one epoch to the next to obtain pseudo observations that compensate for the system dynamics. The Kalman filter is based on the same algorithms as the stepwise adjustment, thus it have the same properties as the LSQ adjustment in terms of linearity, unbiasedness and minimum variance.

The Kalman filter algorithm is recursive and consists of three steps: prediction, gain calculation and update step. The prediction step is the only new input from the sequential adjustment point of view, so it will be studied in the subsequent section, followed by a section with a summary of the complete Kalman filter algorithm.

#### 2.1.3.1 The Prediction Step

The additional model that is used in the Kalman filter is based on a first order differential equation, here given in general form at time $t$ as:

$$\dot{X}(t) = F(t)X(t) + G(t)u(t)$$ (2.17)

$X(t)$ is the state vector of the process, $\dot{X}(t)$ its time derivate, $F(t)$ the system dynamic matrix, $G(t)$ a coefficient matrix to the random forcing function $u(t)$ that usually is assumed to be white noise. Without further information about the parameters in the state vector, they would follow the trajectory given by the differential equation, but if observations are available they could be updated. The relation between the state vector and the observation vector $L$ at time $t$ is:

$$L(t) = H(t)X(t) + v(t)$$ (2.18)

where $H(t)$ is the matrix relating the observations $L(t)$ and their errors $v(t)$ to the state vector $X(t)$.
Eqs. (2.17) and (2.18) express a dynamic continuous-time model of the Kalman filter, so when observations are added at discrete epochs in time \((t = t_0, t_1, ..., t_{k-1}, t_k, ... )\), these equations have to be converted into a discrete form.

Starting with the observation equation of a dynamical system at time \(t_k\) given as:

\[
L_k = H_k X_k + \nu_k
\]  

(2.19)

\(H_k\) is design matrix, which relates the unknown parameters in state vector \(X_k\) to the observation vector \(L_k\) and error vector \(\nu_k\) in epoch \(k\). The observation errors \(\nu_k\) are assumed to be normally distributed with zero mean, just as the observations in the LSQ-adjustment, with a covariance matrix \(R_k\).

The solution of Eq. (2.17) below follows Brown and Hwang (1997). First the dynamic model of the state vector is converted into a discrete model, and thereafter the corresponding covariance matrix. The state space model is used to derive the difference equation relating samples of \(X\).

The solution of Eq. (2.17) at time \(t_k\) (or epoch \(k\)) is transformed into discrete form by using its particular solution:

\[
X_k = T_{k,k-1} X_{k-1} + \int_{k}^{k+1} T_{k,\tau} G \tau u_{\tau} d\tau = T_{k,k-1} X_{k-1} + w_k
\]  

(2.20)

where the transition matrix \(T_{k,k-1}\) describes the non random transition of the state from time \(k - 1\) to \(k\) and the vector \(w_k\) is the driven response containing the integrated white noise during the time interval \(\tau\) between the epochs. The driven response is assured to contain white noise, since it contains the integrated white noise of the random forcing function \(u(t)\).

The covariance matrix \(Q_k\) related to \(w_k\) is assumed to be known with the following properties:

\[
E\{w_k w_i^T\} = \begin{cases} Q_k, & i = k \\ 0, & i \neq k \end{cases}
\]  

(2.21)

where \(i\) and \(k\) are two epochs in time.

The transition matrix \(T\) and the vector \(w_k\) are the only parameters that differs the Kalman filter from sequential adjustment. It is possible to find the relationship to sequential adjustment by setting the transition matrix \(T\) to an identity matrix and the elements in the driven response vector \(w_k\) to zero. This implies that the predicted state vector will have exactly the same values as in the previous epoch, which is the case in sequential adjustment.

The transition matrix \(T\) has some special properties, as it is an identity matrix \(I\) when \(t = t_0\):

\[
T(t, t) = I
\]  

(2.22)

and its time derivate is:
\[ \dot{T}(t, t_0) = F(t)T(t, t_0) \]  

(2.23)

This implies that the transition matrix can be found by numerically solving this differential equation; see further Brown and Hwang (1997 pp.202-204). An alternative method to determine the transmission matrix is by assuming the system is time invariant for short time intervals \( \Delta t = t - t_0 \). In this case the system dynamic matrix \( F \) becomes constant during the time interval \( \Delta t \) and it is possible to express the transition matrix with an exponential function that has the following Taylor expansion:

\[ T(t, t_0) = e^{F \Delta t} = I + \Delta t F + \frac{(\Delta t F)^2}{2!} + \frac{(\Delta t F)^3}{3!} + \ldots \]  

(2.24)

This relationship implies that the correlation in time between two observation epochs will decrease exponentially. Epochs that are close in time will have a high correlation and the opposite if the time interval is high. With short time intervals \( \Delta t \) it becomes possible to use a linear approximation of \( T \) as:

\[ T(t, t_0) \approx I + \Delta t F \]  

(2.25)

To use the predicted state vector it is also necessary to predict the corresponding covariance matrix of the driven noise \( w_k \), which is expressed as:

\[
Q_k(t) = E\{w_k w_k^T\} = T(t, t_0) Q_k(t_0) T(t, t_0)^T + \int_{t_0}^{t} \int_{t_0}^{s} T(t, \tau) G(\tau) E\{u(\tau)u^T(\tau)\} G(s)^T T(s, \tau)^T d\tau ds
\]  

(2.26)

The first part of this equation transfers the covariance matrix from one epoch into the next, and the second part, the double integral; integrate the variance of the driven response between the epochs. \( E\{u(\tau)u^T(\tau)\} \) is a matrix containing the power spectral density (PSD) of the forcing function \( u \), and it is normally denoted as \( Q \). The PSD shows how much the variance of each parameter in the state vector is expected to change during the time between the observations. The recursive equation for the covariance matrix of \( X \) is given by:

\[ Q_{X,k} = T_{k,k-1} Q_{X,k-1} T_{k,k-1}^T + Q_k \]  

(2.27)

Using Eq.(2.24), the solution of the double integral in Eq.(2.26) can be approximated with the following integral, Farrell and Barth (1999, p.86):

12
\[ Q_k(t) = E\{w_k w_k^T\} \]
\[ \approx Q_0 \Delta t + (FQ_G + Q_G F^T) \frac{\Delta t^2}{2} \]
\[ + [F^2 Q_G + 2FQ_G F^T + Q_G (F^T)^2] \frac{\Delta t^3}{6} + \ldots \]  

where the matrix \( Q_G \) is defined as \( Q_G = GQG^T \).

The presented prediction procedure is true under the same assumptions about the independence and distribution of observations that are used in the sequential adjustment. Furthermore, it is assumed that the covariance matrix of the driven response \( w_k \) and the observation error \( v_k \) is uncorrelated as:

\[ E\{w_k v_i^T\} = 0, \quad \text{for all } k \text{ and } i \]  

### 2.1.3.2 Summary of the Kalman filter algorithm

The discrete Kalman filter has a recursive algorithm were each recursive calculation loop consists of three steps. These can be summarised by combining equations of the sequential adjustment with the prediction method presented in the previous section. Before this is made we need to introduce a set of initial values for the state vector and its variance matrix, which has to be reasonably known from the beginning to prevent the Kalman filter to diverge, Brown and Hwang (1997). After the initialisation follows the prediction step which is the first of three steps that are performed for each recursive loop. In this step parameters are predicted from the previous epoch into the actual epoch by using the prediction equations, derived in the previous section. We are justified in ignoring the contribution of \( w_k \) in Eq. (2.20) because it has zero mean and is not correlated with any if the previous \( w \)'s, (Brown and Hwang 1997), thus we have:

\[ \hat{X}_k^- = T_{k,k-1} \hat{X}_{k-1} \]  

and

\[ Q_{\hat{X},k}^- = T_{k,k-1} Q_{\hat{X},k-1} T_{k,k-1}^T + Q_k \]  

In these equations new super scripts are introduced that continuously will be used in this report; “-” symbolises a predicted and “^”an estimated parameter.

After the prediction step follows the gain calculation, which uses the same equations as in the sequential adjustment Eq.(2.14), here repeated for convenience with the Kalman filter matrix notation:

\[ K_k = Q_{\hat{X},k}^- H_k^T [R_k + H_k Q_{\hat{X},k}^- H_k^T]^{-1} \]
Finally the last step in the recursive algorithm is the update given by Eq. (2.13) as:

$$\hat{X}_k = \hat{X}_k^- + K_k[L_k - H_k\hat{X}_k^-]$$  \hspace{1cm} (2.33)

The update of the covariance matrix can be made by using Eq.(2.15), but here a slightly modified equation is used that conserves the symmetry and positive definiteness of the covariance matrix, see (Brown and Hwang 1997 p.347):

$$Q_{x,k} = (I - K_kH_k)Q_{\hat{x},k}^- (I - K_kH_k)^T + K_kR_kK_k^T$$  \hspace{1cm} (2.34)

In each recursive loop the Eqs.(2.30) to (2.34) are repeated. Jekeli (2001) uses Figure 1 to explain the Kalman filter loop. The upper part of the figure shows the state vector loop in a Kalman filter and the lower part of the loop of the covariance matrix. The upper loop starts in the initial state where $\hat{X}_0$ is introduced and the lower where the corresponding covariance matrix $Q_{\hat{x},0}$ is inserted. Once the filter is started it will produce an updated estimate of the state vector and its covariance matrix for each epoch.

![Figure 1. Kalman filter loop (Jekeli 2001)](image-url)
The Kalman filter algorithm assumes the following properties of the white noise processes $\mathbf{w}_k$ and $\mathbf{v}_k$:

\begin{align}
E[\mathbf{v}_k] &= \mathbf{0} \quad (2.35) \\
E[\mathbf{v}_i \mathbf{v}_j^\top] &= Q_i \delta_{ij} \\
E[\mathbf{w}_k] &= \mathbf{0} \quad (2.37) \\
E[\mathbf{w}_i \mathbf{w}_j^\top] &= R_i \delta_{ij} \quad (2.38) \\
E[\mathbf{w}_k \mathbf{v}_i^\top] &= \mathbf{0} \text{ for all } i \text{ and } k \quad (2.39)
\end{align}

where

\begin{align}
\delta_{ij} &= 1; \text{ when } i = j \\
\delta_{ij} &= 0; \text{ when } i \neq j \quad (2.40)
\end{align}

The Kalman filter is designed to be optimal linear estimators in LSQ sense of the unknown parameters up to and including the actual epoch $k$. Thus, it will not give an optimal solution in situations when the relationship between observations and unknown parameters in the state vector, or the solution of the differential equation in the prediction step, is non-linear. In general it is accepted to choose linear models as an approximation. This type of Kalman filter is called an Extended Kalman filter (EKF), and it is probably the most used form of the Kalman filter in positioning algorithms. The update equation for the EKF is of the same form as the linear Kalman filter given in Eq. (2.33), but instead of using the linear relationship given in the design matrix $\mathbf{H}$, the non-linear observation equation $\mathbf{h}$ is used to determine predicted observations using the predicted state vector $\hat{\mathbf{x}}_k^-:

\hat{\mathbf{x}}_k = \hat{\mathbf{x}}_k^- + \mathbf{K}_k [\mathbf{L}_k - \mathbf{h}(\hat{\mathbf{x}}_k^-)] \quad (2.41)

### 2.1.3.3 Quality control

All observations from any measurement procedure contain some kind of measurement errors. These errors will be transferred into the unknown parameters if they are not removed before the LSQ-adjustment. This is true either if it is a post processing scenario, as when a traditional geodetic network is adjusted, or if it is a real-time procedure.

Measurement errors are normally classified into three groups: random, systematic and gross errors. The random errors behave randomly and affect the measurements in a non-systematic way. Consequently, the random errors will be normally distributed with zero mean. The standard deviation of the measurements depends on the actual measurement precision. Systematic errors follow some physical or mathematical rules and influence the surveying result.
systematically with biases. The cause of this kind of errors can be the measurement instruments, physical environment in which the measurements are made, human factors and measurement routines, wrong modelling of the systematic errors etc. The influence of the systematic errors can be reduced and even removed e.g. by careful calibration of instruments, good models of the physical environment and well selected measurement routines. Gross errors are due to human mistakes or malfunctioning instruments. Gross errors do not follow any rules and therefore it is impossible to create some direct mathematical model that removes them.

An approach to remove gross errors was introduced by Baarda (1968). Baarda considered them as large measurement residuals and he used hypothesis testing to find them. This approach is known as data snooping, and it is well documented in both papers and textbooks on LSQ-adjustment of geodetic networks as Bjerhammar (1973), Tunissen (1985) and Kuang (1996), to mention just a few.

Data snooping is a very efficient procedure to find gross errors, but it assumes that there is only one gross error present at the time. Even if it is common to perform iterations when more than one error is present, the method becomes less effective. Furthermore, it is based on the assumption that observations with large gross errors will also have large residuals from a preliminary least squares adjustment. However, many studies have shown that LSQ adjustment has the tendency to smooth residuals, Fan (2003). This implies that it cannot be guaranteed that observations with large gross errors can be found based on the use of residuals.

It is not only the observation accuracy and precision that influence the result in a LSQ-adjustment. The design of the model that connects observations with the unknown parameters will also influence the estimates. The term reliability is often used to describe the quality of estimated parameters with respect to gross and systematic errors. High reliability ensures that it is easy to detect small outliers or gross errors. Reliability is subdivided into internal and external reliability. High internal reliability indicates that small outliers in the observations can be detected and a high external reliability suggests that the influence of an undetected gross error will have a small influence on the estimated parameters.

2.1.3.3.1 Quality control in Kalman filter

The Kalman filter produces an optimal unbiased estimation in least squares sense of the state vector for each epoch based on all observations up to and including the last epoch; see Section 2.1.3.2. However, this is only true as long as the assumptions about the underlying mathematical model hold. This can be checked with a quality control algorithm as the data snooping algorithm is adapted to real-time applications, but as mentioned earlier, it has some limitations, because it
is based on residuals attained from an initial adjustment. This problem could be overcome in a real-time filter by using predicted residuals:

$$v_k = L_k - h(\hat{X}_k^-)$$  \hspace{1cm} (2.42)

Together with the corresponding covariance matrix of the predicted residuals:

$$Q_{v_k} = R_k + H_k Q_{\hat{X}_k^-} H_k^T$$  \hspace{1cm} (2.43)

There are two advantages in using predicted residuals compared with the estimated residuals used in the data snooping algorithm. First, they are available before the gain calculation step in the Kalman filter and they are not correlated with the observations at the actual epoch, according to the assumptions of Eq. (2.29). These properties are very useful, since it is possible to perform the quality control before the new observations enter the Kalman loop. Willsky (1976) have summarised several useful methods to perform quality testing in a Kalman filter. Here we follow the work of Teunissen and Salzmann (1989) and Teunissen (1990).

Real-time quality control can be split into three steps detection, identification and correction. The first two steps can be recognised from the data snooping algorithm. Within the detection step is an overall test performed to diagnose whether an unspecified observation error has occurred or not. If the detection step signals that there is an error then the identification step follows, where the potential sources of the model error are identified by a set of alternative hypothesis. The identification procedure has also a second task, namely to find the starting time of when the error occurred. After identification, the recursive filter has to be adapted to eliminate the state vector biases.

The following hypotheses are considered in the predicted residuals:

$$\begin{align*}
H_0: v_k &\sim N(0, Q_{v_k}) \\
H_A: v_k &\sim N(\nabla v_k, Q_{v_k})
\end{align*}$$  \hspace{1cm} (2.44)

In the zero hypotheses $H_0$ is it assumed that there are no errors in the observations and that the dynamic model as well as the adjustment model is correct. The alternative hypothesis $H_A$ assumes that there are some errors in either the system or in the observations that causes a bias $\nabla v_k$ in the vector of residuals. The bias vector can be parameterised:

$$\nabla v_k = C_{v_k} \nabla_k$$  \hspace{1cm} (2.45)

with a known full rank matrix $C_{v_k}$ and a column vector $\nabla_k$ with the estimated errors (cf. Eq. (2.57)). $C_{v_k}$ is used to specify the alternative hypothesis. How to use $C_{v_k}$ will be described later in this section. The test statistics for testing $H_0$ against $H_A$ is given by Teunissen (1990):
which are tested according to the following hypothesis:

\[
\begin{align*}
H_0: & \quad T_k \sim \chi^2_2(b_k, 0) \\
H_A: & \quad T_k \sim \chi^2_2(b_k, \lambda_k)
\end{align*}
\]  

(2.47)

\(\chi^2_2\) is the upper probability point of the central \(\chi^2\)-distribution with \(b_k\) degrees of freedom and \(\alpha\) the level of significance and

\[
\lambda_k = \nabla_k^T C_{v_k} Q_{v_k}^{-1} C_{v_k} \nabla_k = \nabla_k^T Q_{v_k}^{-1} \nabla_k
\]  

(2.48)

is the non-centrality parameter of the estimated parameter under \(H_A\).

Teunissen and Salzmann (1989) define two types of test levels: local and global. The local test takes observations from one epoch into account, while the global test takes several. Consequently, the global test has a superior detection power compared to the local test, since more observations are involved. This is a good property, but it could also include an unwanted delay in the gross error detection, a delay that is a bit troublesome to implement in a real-time procedure. This is mainly because of the requirement to store several states of the Kalman filter to be able to restart the calculations in any previous epoch. We will only consider the local test in the further work of these theses.

The local overall model is able to detect if there are some gross errors in the model or in the observations, but it cannot identify the error source. It simply makes a binary decision in the hypotheses test to see if there is a gross error or not. This property makes it useful to decide whether to continue with identification on observation level or not. In the overall tests it is the coefficient matrix \(C_{v_k}\) is set to an identity matrix \(I\). The test statistics then become:

\[
T_k = v_k^T Q_{v_k}^{-1} v_k
\]  

(2.49)

and the zero hypothesis will be rejected if and only if:

\[
T_k \geq \chi^2_2(b_k, 0)
\]  

(2.50)

If this hypothesis signals that an error is identified, the next step is to identify the error source. Observation errors can be identified with individual tests at observation level.

In the individual local test, the degree of freedom \(b_k\) is set to 1, which implies that the coefficient matrix \(C_{v_k}\) becomes a vector denoted \(c_{v_k}\) and the vector \(v_k\) becomes a scalar. The individual test statistics is:
\[ t_k = \frac{c_{v_k} Q_{v_k}^{-1} \nu_k}{\sqrt{c_{v_k}^T Q_{v_k}^{-1} c_{v_k}}} \]  \hspace{1cm} (2.51)

where \( c_{v_k} \) is used to choose the alternative hypothesis, as:

\[ c_{v_k} = (0,0,...,1, ... 0)^T \]  \hspace{1cm} (2.52)

This method follows the data snooping approach, see Teunissen (1985). The hypothesis used in the individual one-dimensional test under \( H_0 \) and \( H_A \) are

\[
\begin{align*}
(H_0): & \quad t_k \sim N(0,1) \\
(H_A): & \quad t_k \sim N(\sqrt{\pi}/\sigma, 1)
\end{align*}
\]  \hspace{1cm} (2.53)

and the hypothesis will be rejected if and only if

\[ t_k \geq \chi_{a/2}(0,1) \]  \hspace{1cm} (2.54)

where \( N(0,1) \) corresponds to a normally distributed parameter with zero mean and standard deviation 1 and \( \alpha \) is the level of signification that indicates the risk of rejecting the null hypothesis even if it is correct (Type I error). A rejected hypothesis indicates that there is a gross error in the observation. Jansson (1998) uses an alternative approach to reject gross errors presented by Teunissen (1985), by using the reliability parameter Minimum Detectable Bias (MDB). By specifying the level of significance \( \alpha \) and the power of the statistical test \( (1 - \beta) \) it is possible to determine the minimum detectable error for each observation. Where \( \beta \) being the risk of incorrectly accepting the null hypothesis even if the alternative hypothesis is true (Type II error). By specifying these parameters it is possible to determine the one dimensional non-centrality parameter \( \lambda \) using the same equations that normally is used in hypothesis testing to determine the non-centrality parameter \( \delta \) that corresponds to the smallest shift of the standard normal distribution that is possible to detect. If \( \alpha = 5\% \) and \( \beta = 20\% \) the non-centrality parameter becomes \( \lambda \approx 2.80 \), (Leick 2004 p.156). \( \lambda \) are used in Eq.(2.51) instead of \( c_{v_k} Q_{v_k}^{-1} \nu_k \) to determine MDB as:

\[ |\nu_k| = \sqrt{\frac{\lambda_k}{c_{v_k}^T Q_{v_k}^{-1} c_{v_k}}} \]  \hspace{1cm} (2.55)

which is the minimal detectable error in an observation given \( \alpha \) and \( \beta \). This is easily calculated within the Kalman filter since both the predicted residuals and its covariance matrix is available for each epoch before the gain calculations are started.

The alternative hypothesis is given by comparing \( \nu_k \) with the best estimate of the error:

\[ |\nu_k| \geq \hat{\nu}_k \]  \hspace{1cm} (2.56)

where \( \hat{\nu}_k \) is calculated as:
\( \hat{p}_k = (c^T_{v_k} Q_{v_k}^{-1} c_{v_k})^{-1} c^T_{v_k} Q_{v_k}^{-1} v_k \)  \hspace{1cm} (2.57)

with the covariance matrix:
\[ Q_{v_k} = (c^T_{v_k} Q_{v_k}^{-1} c_{v_k})^{-1} \]  \hspace{1cm} (2.58)

The proposed local identification procedure can quite easily detect errors in the observations larger than the minimal detectable error but when it comes to model errors it is more difficult. The algorithm above is based on the predicted residuals, determined as the difference between actual and predicted observations. Predicted observations are generated with their observation equations and the parameters in the predicted state vector. These predicted parameters will be incorrect if the used dynamic model does not reflect the physical reality. And if some of the parameters are common in several observations, which they usually are, all of them will be influence of the erroneous dynamic model and therefore also the predicted residuals. The predicted residuals are as mentioned earlier used in the gross error detection procedure, and with errors in the dynamic model it becomes difficult to determine if the detected errors are observation errors or model errors. Consequently, it is important to be careful when several gross errors are detected at the same time within the observations, since this could indicate a model error.

The last step in the quality control algorithm is the adoption procedure, where the observations of the model are corrected for the detected gross errors. This will be discussed more in Section 2.4.4 that concerns quality control in the implemented Kalman filter.

### 2.2 Positioning with GPS

In this section we will develop the functional model for the GPS code and phase observables. The main purposes are firstly to identify and relate the unknown parameters within each of the observables, secondly to describe different methods used to remove singularities caused by linear relations among the unknown parameters and finally to compare the methods to motivate the choice of GPS observation equations to be used in the Kalman filter updates.

#### 2.2.1 Observation equations

In general Code pseudorange, phase and Doppler-count are the three different types of observables that are performed with high precision geodetic GPS-receivers. These are collected, as samples of continuous time series processes in epochs with fixed sample intervals, e.g. each second.
The purpose with this section is to derive the observation equations for code and phase observations. The Doppler observations are not used in this thesis and are therefore not described further. The derivation of the observation equations follows the derivations given by de Junge (1998), Hoffmann-Wellenhof et al. (2001) and Leick (2004).

2.2.1.1 Code Pseudorange observations

Code pseudorange observations are based on the PRN-code message modulated on the carrier phase signal. They can be used to determine the travelling time of a signal from satellite to receiver, and thereby also the distance by multiplying the travelling time with the speed of light. The travelling time is determined by the receiving instrument with use of correlation technique. The shape of the PRN-code is a-priori known and its replica is generated by the instrument. Maximum correlation between the replica and the incoming signal is determined by time shifting the generated signal and the total time shift corresponds to the travelling time of the signal from satellite to receiver.

The pseudorange $P^S_A$ between a satellite $S$ and a receiver $A$ can be expressed as:

$$ P^S_A(t_A) = (t_A - t^S)c $$  (2.59)

$t_A$ and $t^S$ are the nominal times of the signal reception in the receiver $A$ and emission from the satellite, respectively, and $c$ represents the speed of light. From now on capital lettered subscripts are used to denote different receivers and the corresponding superscripts the satellites. The transmitted PRN-code is generated in the satellite by the satellite clock and is recorded in the GPS-receivers according to the receiver clock. The nominal times are related to true GPS-times, $t^*_A$ and $t^*_S$, at receiver $A$ and satellite $S$ as:

$$ t^*_A = t_A - \delta t_A $$  (2.60)

and

$$ t^*_S = t^S - \delta t^S $$  (2.61)

where $\delta t_A$ and $\delta t^S$ are introduced as the receiver and satellite clock offsets with respect to true GPS-time. Combining Eqs. (2.59) and (2.60) we get an expression of the pseudorange observation

$$ P^S_A(t_A) = [(t^*_A + \delta t_A) - (t^*_S + \delta t^S)]c $$
$$ = (t^*_A - t^*_S)c + \delta t_A c - \delta t^S c $$
$$ = \rho^S_A(t^*_A - t^*_S, t_A^*) + \delta t_A c - \delta t^S c $$  (2.62)

where the topocentric distance $\rho^S_A(t^*_A - t^*_S, t_A^*)$ marks the true geometric distance travelled by signal emitted by satellite $S$ at time $t^*_A - t^*_S$ and received by receiver $A$ at the true time
instance $t_A$, $r_A^s$ represents the travel time of the signal e.g. the time the signal needs to travel from the satellite to the receiver. The topocentric distance is calculated:

$$
\rho_A^s(t_A^*) = \rho_A^s(t_A^* - t_A^*) = \left\|x^s(t_A^* - t_A^*) - x_A(t_A^*)\right\| = \sqrt{(x^s - x_A)^2 + (y^s - y_A)^2 + (z^s - z_A)^2}
$$

(2.63)

where $x^s = [x^s, y^s, z^s]$ and $x_A = [x_A, y_A, z_A]$ are coordinate vectors with the satellite and the receiver positions given in an Earth Centred Earth Fixed (ECEF) coordinate system. Since the true time $t_A^*$ is unknown it is not possible to directly calculate the topocentric distance. To overcome this are Eq. (2.63) linearised around its nominal receiver time $t_A$, with a first order Taylor series expansion, as:

$$
\rho_A^s(t_A^*) = \rho_A^s(t_a) + \rho_A^s(t_a) \delta t_A
$$

(2.64)

Equation (2.62) can now be rewritten as:

$$
P_A^s(t_A) = \rho_A^s(t_a) + \rho_A^s(t_a) \delta t_A - \delta t^s c
$$

(2.65)

The term $\rho_A^s(t_a) \delta t_A$ in equation (2.65) is often neglected if the equation is used for single point positioning. The reason is that the absolute value of the topocentric range rate $\rho_A^s(t_a)$ is always less than 800 m/s. If the receiver clock correction is $\delta t_A = 10 \mu s$, then $\rho_A^s(t_a) \delta t_A = 8 mm$, this is far below the observation noise level. However, this correction is significant in case of relative positioning with phase pseudoranges.

### 2.2.1.2 Phase observations

When a GPS-receiver is started at nominal time $t_A$, it generates a carrier with a nominal frequency $f_A$ and phase $\varphi_A(t_A)$, based on the receiver clock. Incoming signals from a satellite are created within the satellite clock and reconstructed in the receiver with a carrier frequency $f^s$ and phase $\varphi^s(t_A)$. The difference between the phases is called beat phase:

$$
\varphi_A^s(t_A) = \varphi^s(t_A) - \varphi_A(t_A)
$$

(2.66)

The received phase and generated replica is described by, Hoffman-Wellenhof et al. (2001 p.88), as:

$$
\varphi^s(t_A) = f^s t_A - f^s \frac{\rho_A^s(t_a)}{c} + f^s \delta t^s
$$

(2.67)

and

$$
\varphi_A(t_A) = f_A t_A + f_A \delta t_A
$$

(2.68)

where $\rho_A^s$ is as before the geometrical distance between receiver and satellite, $c$ the speed of light, $\delta t^s$ and $\delta t_A$ are time delays introduced in Eq. (2.60). The frequency difference between $f_A$
and $f^s$ is in the order of a fraction part of Hertz, so they can be assumed to be the equal ($f = f^s = f_A$). According to this, the beat phase can be simplified as:

$$\varphi^s_A(t_A) = -f \frac{\rho^s_A(t_A)}{c} + f(\delta t^s - \delta t_A)$$  \hspace{1cm} (2.69)

The actual observation within a GPS-receiver is formed by the total sum of the fractional beat phase cycles over time from the epoch when the instrument started $t_0$ until the current time $t$. The geometric distance between satellite and receiver in terms of beat phases is composed of the receiver phase observation $\Delta \varphi^s_A|_{t_0}$ and an unknown additional integer number value $N^s_A$:

$$\varphi^s_A(t) = \Delta \varphi^s_A|_{t_0} + N^s_A$$  \hspace{1cm} (2.70)

$N^s_A$ is also known as the phase ambiguity or just ambiguity. This value will remain fixed as long as the instrument is tracking a satellite without interruption. If the signal is interrupted between satellite $S$ and receiver $A$, $N^s_A$ will get a new value, an event also known as a cycle slip, this will be discussed further in Section 2.4.2.3.

Inserting Eq. (2.70) into Eq.(2.69) we get the following expression of the phase observations:

$$-\Delta \varphi^s_A|_{t_0} = f \left( \frac{\rho^s_A(t_A)}{c} \right) - f(\delta t^s - \delta t_A) + N^s_A$$  \hspace{1cm} (2.71)

where the negative part, on the left side of the equal sign, is the receiver observation in epoch $t$. This expression is given in wavelengths but it can easily be converted into units of metres using the physical relation between wavelength and frequency $f = c\lambda^{-1}$:

$$\Phi^s_A(t_A) = -\lambda \Delta \varphi^s_A|_{t_0} = \rho^s_A(t_A) + c \delta t_A - c \delta t^s + \lambda N^s_A$$  \hspace{1cm} (2.72)

The true time $t_A$ is unknown, so to determine an approximate value of the topocentric distance we have linearised the geometric distance around the known nominal receiver time $t_A$ in Eq. (2.64). The value of $\rho^s_A$ is the velocity in geometric range at the true time of the receiver. As mentioned in the previous section, this value is large enough to influence the phase observation. Equation (2.72) can also be expressed in metres as:

$$\Phi^s_A(t_A) = \rho^s_A(t_A) + \left( \frac{\rho^s_A(t_A)}{c} \right) \delta t_A - c \delta t^s + \lambda N^s_A$$  \hspace{1cm} (2.73)

where $\lambda$ is the actual wavelength. This expression of the phase observation equation will be used in the following context when using phase observations.

### 2.2.1.3 Biases and noise

The observation equations for code and phase pseudoranges, derived so far, are valid in the case that no systematic biases or random errors are present except for the clock biases. In reality the
observations are apart from random observation errors also influenced by several systematic biases. Hoffman-Wellenhof (2001) summarises the influencing biases according to their source in three groups: satellite, atmosphere and receiver biases. In Table 1 these groups are presented together with the related biases. The sub and superscripts are used to indicate if a bias is related only to a satellite or a receiver or to a combination of them. Subscript $i$, can be either $L1$ or $L2$, denoting the dependence if the variable on the frequency, and subscript $P1$ and $P2$ denotes code measurement. These subscripts are necessary, since some biases are depending on the actual observation type.

Table 1. Additional biases that affect the GPS observables

<table>
<thead>
<tr>
<th>Satellite</th>
<th>$\delta O^S_A$ - Orbital errors</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$\delta t^S_A$ - Satellite clock offset</td>
</tr>
<tr>
<td></td>
<td>$\delta H^S_A$ - Hardware delay bias in the satellite</td>
</tr>
<tr>
<td></td>
<td>$\delta A^S_i -$ Antenna offset in the satellite</td>
</tr>
<tr>
<td></td>
<td>$T^S_{GD}$ - Satellite Code offset</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Atmosphere</th>
<th>$I^S_A$ - Ionospheric delay</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$T^S_A$ - Zenith Tropospheric delay (wet and dry)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Receiver</th>
<th>$\delta M^S_{A,i}$ - Multipath</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$\delta t_A$ - Receiver clock offset</td>
</tr>
<tr>
<td></td>
<td>$\delta H_{A,i}$ - hardware delay bias in the receiver</td>
</tr>
<tr>
<td></td>
<td>$\delta A^S_{A,i}$ - Antenna phase centre variations</td>
</tr>
</tbody>
</table>

All satellite biases are related to the satellite system e.g. the hardware and software in the satellites, the accuracy in the used ephemerides and the corresponding satellite clock correction model (Precise or Broadcast ephemerides). High accuracy ephemerides reduce the orbital errors $\delta O^S_A$ and similarly an accurate model for the satellite clock error $\delta t^S_A$ reduces its influence on the final result. The standard orbits and clock correction models are found in both the broadcasted and precise ephemerides. The broadcasted ephemerides are a part of the navigation message that is modulated on the carrier phase and is thus always available in the GPS-receivers. These ephemerides contain also some general information about the actual status of the satellites, their orbital information and clock parameters that describe the satellite clock offset: Orbital information for each satellite is described by six parameters for the Kepler ellipse, three secular correction terms and six periodic correction terms. The clock offset correction model is usually given by coefficients for the bias, drift and frequency drift. These values are determined at the GPS Master Control station based on observations collected at all
the monitor stations of the GPS control segment. The accuracy in these orbits is \( \sim 1.6 \text{ m and } \sim 7 \text{ ns} \), (IGS-homepage). The format of the broadcasted ephemerides is described in the document (ICD-GPS-200).

The International GNSS Service (IGS) delivers precise GPS-ephemeris in three different types of orbits: final, rapid and ultra-rapid. The Final orbits are the most accurate of them with an orbit accuracy of <0.05 m and clock accuracy <0.1 ns, but the problem with these orbits, as well as with the Rapid class, is that they are available at 12 days and 17 hours latency, i.e. they are not suitable in real-time applications. Ultra-Rapid IGS orbits on the other hand, contain 48 hours of orbits; the first half computed from observations and the second half predicted orbits, according to IGS product description (IGS-homepage). IGS-ephemerides are distributed through the internet in tabular form, the coordinates of each satellite together with the actual clock correction is given with an interval of 15 minutes. For further information about how to use the IGS-products, see Kouba (2003) or (IGS-homepage). The satellite position and clock accuracy in the Ultra-Rapid orbit are \( \sim 10 \text{ cm and } \sim 5 \text{ ns} \), which are much better than the precision of the broadcasted ephemerides, i.e. also very useful in positioning applications. The need for an internet connection is the only consideration in this approach.

Besides the orbital and the satellite clock offsets the group of satellite biases also contains satellite hardware delays \( \delta H^S \) and an antenna offset \( \delta A^S \) parameters. The satellite hardware error is often assumed to be zero, because they cannot be separated from the clock offset. Consequently the clock offsets will implicitly compensate for the hardware delay, (Scheer 1999).

The antenna offset \( \delta A^S \) is used to describe the offset between the calculated position of a satellite and the actual position of the phase centre of the emitting antenna. The antenna offset is zero for the broadcasted ephemerides and thereby the true position phase centre of the satellite antenna is calculated. However other ephemerides like the one calculated by International GPS Service for Geodynamics (IGS) are defined for the satellite centre of mass and corrections are needed to compensate for mass/phase centre offset, see Warren and Raquet (2003). We have developed an algorithm that is independent of the actual distribution format of the orbits; see Section 2.4.2.

In each satellite there is a time offset between the C1 and P2 code message caused by the satellite hardware and known as the satellite code offset \( (\tau_{\text{OD}}^S) \). These values are inseparable from the receiver hardware delay and are therefore necessary to allow dual-frequency users to conventionally eliminate the ionospheric delay. The \( \tau_{\text{OD}}^S \) values are initially determined during factory testing. In principle the hardware delays are time-dependent, although in practice they seem to be relatively stable, at least on the time scale of days, Scheer (1999) and because of this
are these values routinely estimated by the control segment of the GPS system. The updated values are made available to the user through the broadcast message.

The second group of biases in Table 1 is related to the atmosphere, which consists of the influence of the ionosphere and the troposphere. Ionospheric bias $I_{A,i}^S$ depends on the actual ionospheric conditions along the path. It has a positive influence on code observations and a negative one on phase observations and it is further influenced by the actual signal frequency. The tropospheric bias, $T_A$, is always positive, independent of frequency and depends on the tropospheric conditions along the ray path. Its influence is related to the signal travelled distance through the troposphere, thus observations to satellites with low elevation angle are more influenced than satellites at high elevation angles. This property is used when estimating the tropospheric delay. Instead of estimating one parameter for each satellite/receiver combination, one parameter is estimated at each receiver representing the troposphere at the station. The relationship between the actual zenith delay and the tropospheric influence at some elevation angle is given by a mapping function $m_A^S$. The influence of the atmospheric delay will further be discussed in Section 2.3.3.2.

Receiver dependent biases depend on the actual receiver, antenna and the environment around the station setup point. Similar to the satellite biases, there occurs for each receiver, an unknown clock offset $\delta t_A$ and a hardware delay $\delta H_{A,i}$. These are unknown and do not have any predefined correction models.

Multipath $\delta M_{A,i}^S$ is caused by single or multiple reflections of a signal before it reaches the receiver antenna. The influence multipath depends on several factors like the environment at the observation site, the used receiver antenna, type of signal (code or phase) etc. The multipath is studied further in Section 2.3.4.2.

The last type of receiver bias is caused by the actual antenna. Each antenna type has a unique phase centre according to the signal frequency, and if this is not handled correctly it will cause biases in the observations; see further Section 2.3.3.2.2.

Adding the systematic errors to Eqs. (2.65) and (2.73) we get the following observation equations for codes:

$$
P_{A,1}^S(t_A) = \rho_A^S(t_A) + (\rho_A^S + \epsilon) \delta t_A - c(\delta t^S - T_{GD}^S) + I_A^S(t_A) + m_A^S T_A(t_A) + \delta O_A^S(t_A) + \delta M_{A,P1}^S(t_A) + \delta H_{A,P1}^S(t_A) + \delta H_{P1}^S(t_A) + \delta A_{A,P1}^S(t_A) + \delta A_{P1}^S(t_A) + \epsilon_{P1}
$$

(2.74)

and
\[
\begin{align*}
P^S_{A,t2}(t_A) &= \rho^S_A(t_A) + (\rho^S_A + c)\delta t_A - c(\delta t^S - \alpha_t T^S_{GD}) + \alpha_t I^S(t_A) + m^S(t_A) + \delta \theta(t_A) + \delta M^S_{A,P2}(t_A) + \delta H^S_{A,P2}(t_A) + \epsilon_{p2} \\
&+ \delta A^S_{A,P2}(t_A) + \delta A^S_{P2}(t_A) + \epsilon_{p2} 
\end{align*}
\]

and phase observables:

\[
\begin{align*}
\Phi^S_{A,t1}(t_A) &= \rho^S_A(t_A) + (\rho^S_A + c)\delta t_A - c(\delta t^S - \lambda_{L1} N^S_{A,L1} - l^S_{L1}) + m^S(t_A) + \delta \theta(t_A) + \delta M^S_{A,L1}(t_A) + \delta H^S_{A,L1}(t_A) + \delta H^S_{L1}(t_A) \\
&+ \delta A^S_{A,L1}(t_A) + \delta A^S_{L1}(t_A) + \epsilon_{L1} \\
\Phi^S_{A,t2}(t_A) &= \rho^S_A(t_A) + (\rho^S_A + c)\delta t_A - c(\delta t^S - \lambda_{L2} N^S_{A,L2} - \alpha_t I^S(t_A) + m^S(t_A) + \delta \theta(t_A) + \delta M^S_{A,L2}(t_A) + \delta H^S_{A,L2}(t_A) + \delta H^S_{L2}(t_A) \\
&+ \delta A^S_{A,L2}(t_A) + \delta A^S_{L2}(t_A) + \epsilon_{L2} 
\end{align*}
\]

where the constant \( \alpha_t = (f_1/f_2)^2 = (77/60)^2 \approx 1.647 \) is introduced to describe the relationship between the frequencies (denoted by the second sub index).

At the end of each of these observation equations it is important to add the random noise \( \epsilon_i \) that follows a general rule of thumb, which says, “The noise level of the pseudo observations is approximately 1% of the signal chip length”, (Seeber 1993 p. 310). This implies that the noise level is about 3 m for the C/A code observations, 0.3 m for P-code observations and about 0.002 m for phase observations. Since the chip lengths are approximately 300, 30 and 0.2 metres.

The influence of some of the observation biases can be reduced with use of models. For example, the influence of the satellite clock is reduced by using parameters that are distributed in the broadcasted navigation message, or the influence of the ionosphere can be reduced by using the ionospheric model. Corresponding models can also be applied to the antenna offsets, the troposphere and to some extend also to multipath. These models are called deterministic models in this thesis; see Section 2.3.

### 2.2.2 Positioning methods

To reach millimetre accuracy with GPS it is necessary to use phase observations, find the correct value of the unknown ambiguities \( N^S_A \) in each observation equations (2.76) and (2.77), and correct for the systematic errors. Usually the LSQ-adjustment, described in Section 2.1.1, is used to estimate the unknown parameters. One of the requirements LSQ-adjustment is that no linear relationships are allowed between the estimated parameters, since then the solution becomes singular. The problem is that some of the parameters in the introduced observation equations are linearly dependent e.g. the ambiguities are linearly dependent with the clock errors and the hardware delays in the dual-frequency case, and in a single-frequency case the ionosphere bias...
is also linearly dependent on these parameters. Over the years two different approaches of relative GPS positioning have been introduced: the differenced and the undifferenced approach. These are introduced and compared in the following sections starting with the differenced.

### 2.2.2.1 Phase differences

Using differenced observations is the most common approach to remove the singularity in observation equations. The basic idea is to use four observations from two receivers towards two common satellites and then subtract them from each other to eliminate all common parameters. In theory this is equivalent to modelling or estimating the eliminated parameters, such that an independent unknown is introduced for every measurement epoch, (Wübbena and Willgalis 2001). This is made in two steps: in the first, two single differences (SD) are created, and these are then combined in a double difference (DD) observation. This procedure is studied in detail, starting with the single differences, in the following subsections.

#### 2.2.2.1.1 Single differences

If we assume that we have simultaneous observations from two receivers (k and m) towards the same satellite p during an epoch t as in Figure 2. Then the undifferenced observation equations are combined into one equation for the SD as:

\[
\Phi_{km}^p(t) = \Phi_k^p(t) - \Phi_m^p(t) \tag{2.78}
\]

By inserting the observation equations from Eq.(2.76) for receivers k and m into Eq.(2.78) we can obtain the following expression for the single difference:

\[
\Phi_{km}^p(t) = \rho_{km}^p(t) + \left(\rho_{km}^p(t) + \epsilon\right) \delta t_{km} + \lambda_i N_{km}(t) + p_{km,i}(t) \\
+ T_{km}^p(t) + \delta O_{km}(t) + \delta M_{km,i}(t) + \delta H_{km,i}(t) \\
+ \delta A_{km,i}^p(t) + \delta A_{km,i}^p(t) + \epsilon_{km,i} \tag{2.79}
\]

In SD the influence of systematic satellite clock errors in the common satellite p are eliminated and the influence of the atmosphere is greatly reduced. The remaining bias parameters are related to the atmosphere and the receivers, which are all denoted with subscript km. These can easily be determined, i.e. the SD combination of the geometric distance

\[
\rho_{km}^p(t) = \rho_k^p(t) - \rho_m^p(t) \tag{2.80}
\]

All of the SD parameters in Eq(2.79) can be formed in a similar way. A set of n-1 SD can be formed with n observations.
Some generalisations of the SD equations are possible. Orbital errors influence the geometric distance between a receiver and a satellite. If the receivers are placed close to each other, then the angle between the two vectors (α in Figure 2) will be small and the effect of orbital error will become much the same at both stations. A large part of the orbital error cancels by single differencing, according to Baueršíma (1983). Its influence is less than 0.1 ppm in the estimated distance between the stations is less than 100 km if precise orbits are used. Removing the satellite antenna offset is another possible simplification when broadcast ephemerides are used. This is allowed since the broadcast ephemerides are determined for the satellite antenna centre; however it is not possible to do the same without bias when using IGS-precise orbits since these are determined for the satellite mass centre instead of the antenna centre, for further information see Warren and Raquet (2003).

With these simplifications the SD equation becomes:

\[
\Phi_{km}^p(t) = \rho_{km}^p(t) + (\dot{\rho}_{km}^p(t) + \epsilon)\delta t_{km} + \lambda_i N_{km}^p(t) + \Gamma_{km,i}^p(t) + T_{km}^p(t) \\
\quad + \delta M_{km,i}^p(t) + \delta H_{km,i}(t) + \delta A_{km,i}^p(t) + \epsilon_{km,i}
\]  

(2.81)

If for some reason any parts of the removed parameters are left, they are assumed to be normally distributed and thereby treated as observation errors in the single differences.
2.2.2.1.2 Double differences

Double differences are generated by combining two single differences from a pair of receivers to two satellites as in Eq. (2.82), where the single differences between receiver \( k \) and \( m \) to satellite \( q \) and \( p \) are combined into a double difference:

\[
\phi_{km}^{pq}(t) = \phi_k^{pq}(t) - \phi_m^{pq}(t)
\]  

which also can be written with the use of Eq. (2.81) as:

\[
\phi_{km}^{pq}(t) = \rho_{km}^{pk}(t) + \lambda_i h_{km}^{pq}(t) + \eta_{km}^{pq}(t) + \gamma_{km}^{pq}(t) + \delta M_{km,i}^{pq}(t) + \delta \Lambda_{km,i}^{pq}(t) + \epsilon_{km,i}^{pq}
\]  

Figure 3 shows an overview figure of a double difference combination.

![Figure 3. Double differences where two single differences are combined](image)

The double differencing procedure eliminates all receiver dependent errors; both the hardware delays and the clock offsets are removed since they occur in both single differences. But the site dependent errors, that are unique at each observation site, are not removed by the double differenced observation equation. The remaining double differenced variables in the double difference equation are determined in a similar manner as in the case of single differences:

\[
\rho_{km}^{pq}(t) = \rho_{km}^{pq}(t) - \rho_{km}^{pq}(t)
\]  

where the new subscript for the parameters indicates the receiver combination and the superscript the satellite combination.
The singularity in the original observation equation is removed in the double differences, but correlations between the observations are introduced. This is the subject of the following sections.

2.2.2.1.3 Correlations

In the previous section we eliminated the singularity in the phase observables mathematically by combining observations from two receivers. The benefit of this procedure is that the unknown parameters related to clocks and hardware in both satellites and receivers are cancelled out. To get a complete overview of the result when differencing it is also necessary to study the stochastic part of the double differences. Let us denote the phase observations with an arbitrary receiver $R$ to the three satellites $o$, $p$ and $q$ as:

$$\Phi(t) = \begin{bmatrix} \Phi^o_R(t) & \Phi^p_R(t) & \Phi^q_R(t) \end{bmatrix}^T$$ (2.85)

The superscript $T$ denotes the transpose of the vector. Each phase observation error is assumed to be normally distributed with a zero expectation value and variance $\sigma^2$. The observation variance depends on the error sources along the signal path. It depends on the travelling distance through the atmosphere and therefore it is directly correlated with the elevation angle. At low elevation angles the distances through the atmosphere are longer than at high elevation angles. The stochastic nature of these observations will be studied further in Section 2.4.2.2. For the moment it is assumed that all observations are uncorrelated with equal variance $\sigma_0^2$. The pure mathematical correlation matrix can therefore be written as:

$$C_\Phi(t) = \sigma_0^2 I$$ (2.86)

$I$ is a unit matrix. With the use of the, single and double difference observation equations (2.81) and (2.83). We can derive the correlation matrices for each type of difference. How this is achieved is described in the following subsections.

2.2.2.1.4 Correlations in single differences

As mentioned earlier single differences (SD) are generated as a combination of observations from two receivers towards one satellite as in Eq. (2.81). When combining two observations into a SD observation mathematical correlations are introduced. The influence is studied with the following example: we use observations from two receivers ($k$ and $m$) towards three satellites ($o$, $p$ and $q$). Their observations are given in two vectors, one for each receiver, as in Eq.(2.85). To simplify the notation we ignore the time index $t$ and write the single difference combination $\Phi_{SD}$ in matrix notation as:

$$\Phi_{SD} = A_{SD} \Phi$$ (2.87)
where the matrices are

\[
\Phi_{SD} = \begin{bmatrix}
\Phi^o_{km} \\
\Phi^p_{km} \\
\Phi^q_{km}
\end{bmatrix}
\]

\[
A_{SD} = \begin{bmatrix}
1 & 0 & 0 & -1 & 0 & 0 \\
0 & 1 & 0 & 0 & -1 & 0 \\
0 & 0 & 1 & 0 & 0 & -1
\end{bmatrix}
\]

\[
\Phi = \begin{bmatrix}
\Phi^o_k(t) & \Phi^P_k(t) & \Phi^q_k(t) & \Phi^o_m(t) & \Phi^P_m(t) & \Phi^q_m(t)
\end{bmatrix}^T
\]

\(A_{SD}\) is the design matrix of the single difference, is also used to derive the stochastic properties of the SD using the law of error propagation of covariance:

\[
C_{SD} = A_{SD} C_{\Phi} A_{SD}^T
\]  

(2.88)

\(C_{SD}\) is the covariance matrix of the three single differences. With uncorrelated observations this becomes the covariance matrix of single differences:

\[
C_{SD} = A_{SD} (C_{\Phi}) A_{SD}^T = \sigma^2 \Phi A_{SD} A_{SD}^T = 2\sigma^2 I
\]  

(2.89)

Hence, there are no correlations between single differences. The size of the matrix \(I\) depends on the number of single differences that can be formed, or, in other words, the number of common satellites at receiver \(k\) and \(m\). Already here it is clear that the only usable observations in the differencing approach are those that are common at least to two receivers. Eq.(2.89) is a general expression for the covariance matrix of single differences when two receivers are used.

2.2.2.1.5 Single differences in a multi-station solution

Assuming that observations are recorded at two or more receivers simultaneously from a common satellite, a new situation is attained, which is a little bit more complicated from the point of view of correlation than when only two receivers were used. To generate single differences it is common to choose one receiver as a reference receiver in the process, from which all observations are included in generated single differences. This procedure will mathematically introduce new correlations among the generated single differences. That is, observations are simultaneously collected at three stations \((k, l\) and \(m)\) from one satellite, see Figure 4.
Figure 4. Three receivers’ \((k, l, \text{ and } m)\) record observations within the same epoch \(t\) to one common satellite \(p\)

Two single differences can be generated with the three observations. It is usual to choose one satellite and one receiver as a reference pair. When choosing the reference satellite, it is preferable to choose one with a high elevation angle, since the influence of an atmospheric error source, like the ionosphere and troposphere is at a minimum. It can be anticipated that in most cases the risk of losing the connection to the satellite, caused by buildings and natural objects, is lower at high elevation angles than at a low elevation.

In this example satellite \(p\) and receiver \(k\) are used as the reference pair and the following single differences are obtained:

\[
\Phi_{SD,MS} = \begin{bmatrix} \Phi^p_{kl} \\ \Phi^p_{km} \end{bmatrix} = A_{SD,MP} \Phi = \begin{bmatrix} 1 & -1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} \phi^p_k \\ \phi^p_l \\ \phi^p_m \end{bmatrix}
\] (2.90)

with the additional subscript \(MS\) indicating a multi-station solution. The correlations are brought forward by the covariance matrix calculated as:

\[
C_{SD,MS} = A_{SD,MS}(\sigma^2_o I)A^T_{SD,MS} = \sigma^2_o \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}
\] (2.91)

As can be seen, correlations are introduced between the observations at single difference level in a multi-station solution.

2.2.2.1.6 Correlations in double differences

To derive the correlation in the double differences we follow the same steps as we did with the single differences and begin with the case where only two receivers are used. The general matrix expression of the double differences can be formed as:
\[
\Phi_{DD} = A_{DD} \Phi_{SD}
\]

(2.92)

Where \( \Phi_{DD} \) is the vector with double differences and \( A_{DD} \) the design matrix which describes the linear combination of the single differences in \( \Phi_{SD} \). Usually it is the single difference that contains the reference pair used as a reference when the double differences are generated. Using the result from the single difference example, in the previous section, we can write the double differences as:

\[
\Phi_{DD} = \begin{bmatrix}
\Phi_{km}^p \\
\Phi_{km}^q
\end{bmatrix} = \begin{bmatrix}
1 & -1 & 0 \\
1 & 0 & -1
\end{bmatrix} \begin{bmatrix}
\Phi_{km}^p \\
\Phi_{km}^q
\end{bmatrix}
\]

(2.93)

In this case the single difference \( \Phi_{km}^p \) is used as the reference when generating the double differences. The covariance matrix of the DD can be derived, with the law of error propagation of covariance's, in the same manner as in the case of single differences:

\[
C_{DD} = A_{DD} C_{SD} A_{DD}^T = \sigma_0^2 \begin{bmatrix}
2 & 1 \\
1 & 2
\end{bmatrix}
\]

(2.94)

The correlations between the observations in the double differences are evident when studying the off-diagonal elements in the matrix in Eq. (2.94).

### 2.2.2.1.7 Double differences in a multi-station solution

The correlation in the double differences shows the same pattern as the multi-station single differences. This pattern is introduced when using one common receiver and satellite pair. The correlation between the double differenced observations will be even more complicated in the multi-station solution. This is obvious when taking the multi-station correlation in Eq. (2.91) and the double difference correlation in Eq. (2.94) into account. We are not going to study the double differenced multi-station covariance matrices here, but instead we give a reference to Beutler et al. (1986) and Beutler et al. (1987). These papers describe, among other things, the derivation of the multi-station covariance matrix in the normal case when all receivers record observations towards the same satellites and how to proceed when exceptions to the normal occur.

Beutler et al. (1986) and (1987) use the same assumption as we did earlier in Eq. (2.86) that all observations have the same variance and that they are independent. This assumption is not completely true when using real observations, since the observation accuracy is influenced by the disturbance along its path through the atmosphere. Thereby, observations towards satellites at high elevation have in general minimal variance. The accepted approach to create double differences with minimum variance is to select a reference satellite at a high elevation.

Furthermore, it cannot be assumed that common satellites are available in each epoch at all receivers. There might be obstacles that are blocking the signal to some of the receivers but not
on the others. In the worst case observations to a satellite are only collected at one receiver. In this case, the observation must be ignored in the double difference approach since it is impossible to generate any double differences within this scenario.

With a change in the observation scheme, the covariance matrix must be redefined. If another satellite other than the reference is lost, it is quite simple to adjust the covariance matrix to the new situation, but in the case that the reference receiver is lost, it is necessary to create a completely new covariance matrix based on another reference satellite. In the two receiver case it is rather easy to adjust the covariance matrix, but in a multi-station solution it becomes more complicated because of the in-between receivers correlation.

2.2.2.1.8 Correlation in time

So far, the correlation has only been studied within one epoch. Usually more than one epoch is used when positions are determined with GPS. To connect them in time it is a requirement to construct a covariance matrix for all observations. In many applications it can be assumed that there is no correlation between the epochs. This gives the following shape of the covariance matrix where all of the off-diagonal elements (matrices) are set to zero:

\[
C = \begin{bmatrix}
C(t_1) & 0 & \cdots & 0 \\
0 & C(t_2) & \cdots & 0 \\
\vdots & \ddots & \ddots & \vdots \\
0 & 0 & \cdots & C(t_n)
\end{bmatrix}
\]  

(2.95)

where \(t_1, t_2, \ldots, t_n\) are the observation times. This assumption is true if all the error sources in the observation equations are eliminated in the differencing procedure each epoch. Some of them are completely eliminated if the observations are made exactly at the same time (the clock errors and the hardware delays), but there are still some parameters which are not, such as the troposphere, ionosphere, multipath and antenna phase centre variations. These parameters are not eliminated since they are unique to each receiver. Deterministic models for each of them have been developed to reduce as much as possible of their influence as described in Section 2.3, but the models are not completely correct as there will still be some errors left after they are applied to the observation equations. It is the time correlated part of the remaining errors that makes the model in equation (2.95) incorrect.

2.2.2.2 Undifferenced solution

There is a linear dependency in the phase observation equation between the ionosphere, the receiver and satellite clock and hardware delay that makes a LSQ-solution singular. One way to eliminate these dependencies is to use the double differenced solution method described above, but there is also an alternative method based on raw phase observations called undifferenced
approach, where all unknown parameters in the observation equation are estimated separately at each epoch in a state vector instead of eliminating them using several observations.

The Kalman filter allows estimated parameters to vary in time according to some first order differential equation that describes the time process of each estimated parameter. These properties make the Kalman filter useful when undifferenced observations are used. The unknown parameters are time dependent e.g. the tropospheric delay can be modelled by some kind of smooth stochastic process in time, since it would not make large jumps from epoch to epoch; see Figure 5.

The number of unknowns in the state vector $X$ will depend on: the number of used receivers, the number of modelled parameters and the number of observations that are recorded at each receiver. It is fairly easy to add new parameters in the state vector when new observations occur and remove them if they vanish. A more detailed description of this approach will be given in Sections 2.3, but in general one can say that several receivers can easily be used together in a multi-station solution.

![Figure 5. The parameters in the observation equations are normally changing smoothly during time like the dashed line in the figure and do not make any sudden jumps](image)

2.2.2.3 Double Differences vs. Undifferenced data

The two most important GPS observables described in the previous sections are undifferenced observations and double differenced observations. In this section we compare the observability in discussing their advantages and disadvantages.

In the double differences several observations are used to completely eliminate the unknown clock errors and hardware delays in satellites and receivers. In theory this procedure is equivalent to modelling or estimating the eliminated parameters, such that an independent unknown is introduced for every measurement epoch. This implies that the eliminated parameters are in principle modelled as a white noise with infinite variance. Thus, it is impossible to use any models to describe how these parameters are changing over time.
The undifferenced approach, on the other hand, uses undifferenced observables and therefore it is necessary to model and estimate all unknown parameters in the observation equation, including the clock and hardware delay parameters. Modelling the dynamic process of these parameters, instead of eliminating them, gives a clear gain of information that can be used to strengthen the final solution.

Not all parameters in the original observation equations are removed in the double differencing procedure e.g. the site dependent parameters, as the multipath and antenna parameters, and the atmospheric influence will not be removed. This can be seen in Eq. (2.83), that is repeated here for convenience:

$$\Phi_{km}^{pq}(t) = \rho_{km}^{pq}(t) + \lambda_i N_{km}^{pq}(t) + \eta_{km,\lambda}^{pq}(t) + \delta M_{km,\lambda}^{pq}(t) + \delta A_{km,\lambda}^{pq}(t) + e_{km,\lambda}^{pq}$$

The absolute nature of these error effects in the original observation are thereby destroyed, and the remaining parameters are the differenced ones. Modelling these differenced parameters is more difficult than modelling the individual effects in the original observations as been done in the undifferenced solution. Instead of modelling these parameters it is often assumed that the differenced parameters are completely cancelled. With these assumptions the covariance matrix in Eq. (2.95) becomes correct, but it represents only a simplification of the real situation. It should be noted that if the receivers are placed within a few kilometres from each other and at the same height, then the influence of the remaining ionosphere and the troposphere parameters will become very small and therefore ignorable, but the influence of the site dependent parameters are not the same at two different observation sites and therefore it is not correct to say that they are eliminated. In the undifferenced approach, all unknown parameters in the observation equations are estimated at each epoch by a Kalman filter. This approach makes it possible to study each unknown parameter in time and apply outlier detection algorithms to detect abrupt shifts in the result. It is important, in this approach, to model the time correlation of each parameter correctly so it represents the true stochastic nature of the estimated parameters. This is one of the main challenges with the undifferenced approach.

A problem with the double differencing procedure is that only observations that are observed at two stations simultaneously can be used. All observations towards satellites that are visible only at one receiver are lost since they cannot be used to form either single or double differences. This problem do not exists in the undifferenced approach, all observations can be used. There are no limitations as in the other approach where the same observations have to be collected at the receivers.

A further problem with the differencing process is the complexity of rearranging the covariance matrix, especially in the multi-station situation, when the number of observations is changing at
the receivers from epoch to epoch. This procedure is, according to Beutler et al. (1987), quite time consuming and thereby makes this approach less attractive in real-time applications. The undifferenced approach offers a solution without the complex correlations between the observations, (cf. Section 2.4), which are introduced in the differenced approach. Since one of the goals of this project is to use several reference receivers in a multi-station solution, and considering the benefits of the undifferenced approach, we will continue to study how a general model for this approach can be formulated in details.

2.3 State vector models

The undifferenced approach offers a simple model for positioning where the number of simplifications and assumptions about the correlation between observations and in time is minimised. In this section the complete observation model for the undifferenced positioning approach is introduced and the following concerns the practical implementation of the model.

The undifferenced approach is based on a Kalman filter where the unknown parameters are estimated each epoch in a state-vector. The time between two observation epochs are denoted with \( \Delta t \) in the following sections. Not all parameters in the observation equations are estimated in the Kalman filter. Some of them are assumed to be known from the beginning e.g. the satellite coordinates and the clock parameters. Others are assumed to have a deterministic and a stochastic part, e.g. the atmospheric and the receiver clock parameter. The deterministic part is removed with use of predefined empirically determined models and the remaining is considered as the stochastic and estimated in the state-vector of a Kalman filter. The types of parameters that are estimated in the receivers are not the same at reference and rover receivers. The coordinates and velocities at the reference receivers are not estimated since these are assumed to be fixed. Instead corrections for the influence of the atmosphere are estimated together with the orbital error and the receiver clock correction. The rover stations are assumed to be in motion and therefore the position and velocity are estimated together with the receiver clock error and the common error. The atmospheric corrections are not estimated at the rover stations. Instead these corrections are interpolated from the values estimated at the reference stations.

In the following subsections the deterministic and stochastic models of each unknown parameter in the observation equation are studied. Starting with the position and velocity of the rover antenna, the receiver clocks, the atmospheric delays, the receiver antenna, the multipath and finally the common errors. The algorithms which determine the satellite positions and the satellite clock corrections are not discussed here; these are introduced in the pre-processing step of the Kalman filter in Section 2.4.2.
2.3.1 Position and velocity

Both for relative and single point positioning, it is necessary to compute the geometric distance between satellite and receiver, which is given by the following equation:

\[
\rho_s^A = \sqrt{(X^s - X_{A,0})^2 + (Y^s - Y_{A,0})^2 + (Z^s - Z_{A,0})^2}
\]  

(2.96)

where \((X^s, Y^s, Z^s)\) are coordinates of satellite \(S\) at time of emission and \(X_{A,0}\) and \(Y_{A,0}\) are receiver coordinates corrected for the Earth rotation during the signal travel time \(\Delta t^s\):

\[
\begin{align*}
X_{A,e} &= X_A - \omega_e Y_A \Delta t^s \\
Y_{A,e} &= Y_A + \omega_e X_A \Delta t^s
\end{align*}
\]

(2.97)

To be able to use standard linear least squares adjustment, Eq. (2.96) has to be linearised. This is performed with a first order Taylor series as:

\[
\rho^s_A(t_A) = \rho^s_{A,0}(t_A) + \frac{\partial \rho^s_{A,0}(t_A)}{\partial X_{A,0}} \Delta X + \frac{\partial \rho^s_{A,0}(t_A)}{\partial Y_{A,0}} \Delta Y + \frac{\partial \rho^s_{A,0}(t_A)}{\partial Z_{A,0}} \Delta Z
\]

(2.98)

where \(\Delta X\), \(\Delta Y\) and \(\Delta Z\) are the differences between approximate coordinates \(X_{A,0}\), \(Y_{A,0}\), \(Z_{A,0}\) and the true coordinates. Partial derivatives take the following form:

\[
\begin{align*}
\frac{\partial \rho^s_{A,0}(t_A)}{\partial X_{A,0}} &= a_x = \frac{X^s - X_{A,0} + Y_{A,0} \omega_e \Delta t - X_{A,0} \omega_e^2 \Delta t^2}{\rho^s_{A,0}} \\
\frac{\partial \rho^s_{A,0}(t_A)}{\partial Y_{A,0}} &= a_y = \frac{Y^s - Y_{A,0} - X_{A,0} \omega_e \Delta t - Y_{A,0} \omega_e^2 \Delta t^2}{\rho^s_{A,0}} \\
\frac{\partial \rho^s_{A,0}(t_A)}{\partial Z_{A,0}} &= a_z = \frac{Z^s - Z_{A,0}}{\rho^s_{A,0}}
\end{align*}
\]

(2.99)

In our monitoring application, it is reasonable to assume that the points are moving slowly. The motion can be modelled by position-velocity model, where constant velocity is assumed. In the case of the position-velocity model (PV-model) we assume that the GPS-antenna is moving with a constant velocity and that the velocity vector is changing randomly, i.e. the velocity is modelled as a random-walk process. This yields the state vector for one rover receiver \(A\):

\[
X_{PV,A} = [X_A \quad v_x \quad Y_A \quad Z_A \quad v_{xA} \quad v_{yA} \quad v_{ZA}]^T
\]

(2.100)

with the dynamic model:

\[
\begin{align*}
\dot{X}_A &= v_x \\
\dot{v}_x &= 0 + u_x
\end{align*}
\]

(2.101)

and the covariance matrix:

\[
E[u_x(t)u_x^T(t)] = Q_A = \begin{bmatrix}
q_{AX} & 0 & 0 \\
0 & q_{AY} & 0 \\
0 & 0 & q_{AZ}
\end{bmatrix}
\]

(2.102)
where the \( q_{AX}, q_{AY} \) and \( q_{AZ} \) are the power spectral density (PSD) of the acceleration with unit \((m^2/s^4)/Hz\). (Jekeli 2001, p.227).

The dynamic matrix \( F_{PV,A} \) for the PV-model, the coefficient matrix \( G_{PV,A} \) and the random forcing function \( u_{a,A} \) are:

\[
F_{PV,A} = \begin{bmatrix}
0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 \\
\end{bmatrix}, \quad G_{PV,A} = \begin{bmatrix}
0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 \\
\end{bmatrix}, \quad u_{a,A} = \begin{bmatrix}
u_{ax} \\
u_{ay} \\
u_{az} \end{bmatrix}
\]

(2.103)

Please note, that we estimate position and velocity only for rover receivers since these receivers are assumed to be in motion. The coordinates at the reference receivers are held fixed. There should always be at least one reference receiver in the network.

Since, in the case of our PV-model, \( F^n = 0, n \geq 2 \), the process noise covariance matrix becomes exactly:

\[
Q_A = Q_0 + \left( FQ_g + Q_g F^T \right) \frac{\Delta t^2}{2} + FQ_g F^T \frac{\Delta t^3}{3}
\]

(2.104)

with

\[
Q_A = \begin{bmatrix}
\left( \frac{q_{ax} \Delta t^2}{3} \right) & 0 & 0 & \left( \frac{q_{ax} \Delta t^2}{2} \right) \\
0 & \left( \frac{q_{ay} \Delta t^2}{3} \right) & 0 & \left( \frac{q_{ay} \Delta t^2}{2} \right) \\
0 & 0 & \left( \frac{q_{az} \Delta t^2}{3} \right) & 0 \\
\left( \frac{q_{ax} \Delta t^2}{2} \right) & 0 & q_{ax} \Delta t & 0 \\
0 & \left( \frac{q_{ay} \Delta t^2}{2} \right) & 0 & q_{ay} \Delta t \\
0 & 0 & \left( \frac{q_{az} \Delta t^2}{2} \right) & q_{az} \Delta t \\
\end{bmatrix}
\]

(2.105)

and the transition matrix for position and velocity is given by:

\[
T_{PV,A} = I + F_{PV,A} \Delta t = \begin{bmatrix}
1 & 0 & 0 & \Delta t & 0 & 0 \\
0 & 1 & 0 & 0 & \Delta t & 0 \\
0 & 0 & 1 & 0 & 0 & \Delta t \\
0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 \\
\end{bmatrix}
\]

(2.106)

where \( \Delta t \) is the time interval between two epochs.
An alternative to the PV-model is the static P-model, where it is assumed that the coordinates of the position of the rover receiver are constant. The unknown parameters in the state vector of this model are given by:

$$X_{P,A} = [X_A \ Y_A \ Z_A]^T$$  \hspace{1cm} (2.107)

with the dynamic model:

$$\dot{X}_A = 0$$ \hspace{1cm} (2.108)

and the covariance matrix:

$$E[u_a(s)u_a^T(t)] = Q_A = \begin{bmatrix} q_{AX} & 0 & 0 \\ 0 & q_{AY} & 0 \\ 0 & 0 & q_{AZ} \end{bmatrix}$$ \hspace{1cm} (2.109)

The dynamic matrix $T_{P,A}$ and the coefficient matrix $G_{P,A}$ in this model are identity matrices.

### 2.3.2 Receiver clock

The nominal time $t_A$ in receiver A is related to true GPS-time as described in Eq.(2.110). The clock delay $\delta t_A$ is not constant in time; it develops as the integral of the frequency error of the clock oscillator. The total clock delay can be described as a bias $\delta t_{b,A}$ plus a drift parameter $\delta t_{dr,A}$:

$$\delta t_A = \delta t_{b,A} + \Delta t \delta t_{dr,A}$$ \hspace{1cm} (2.110)

How the bias and drift values vary in time depends on the receiver design. Leica receivers correct the receiver clock instantly each epoch to remove the drift parameter. Another approach is used in Javad and Trimble receivers; they allow the clocks to drift until the total drift is 1 millisecond, then the receiver adjusts the clock. The drift rate is not the same in the receivers not even within the same brand, for instance see Figure 6 where the clock drift in two Javad Lexon-GGD receivers are compared with the clock drift of a Trimble R7 receiver.

To avoid different receiver design specified algorithms to estimate the influence of the receiver clocks is one parameter $\delta t_{A,SPP}$ estimated for both the drift and the bias. This parameter is treated as the deterministic part of the receiver clocks and it is performed with a Single Point Positioning (SPP) algorithm, see forward Hoffmann-Wellenhof et al. (2001, p.257-259). The SPP algorithm is designed for positioning and estimates not only the unknown coordinates but also the unknown receiver clock and the influence of the ionosphere when dual frequency observations are used. In our case the coordinates at the reference and rover stations are already known. The coordinates at the reference stations are known from the beginning and held fixed and the coordinates of the roving receivers are estimated in the main positioning.
algorithm, explained in Section 2.4. These values are used to improve the parameter estimation in the SPP algorithm by reducing the number of unknown parameters. The 1ms corrections (jumps) are counted within the algorithm at each receiver. The estimated clock corrections are used as deterministic parameters and are therefore moved to the left side in the observation equation together with the total number of 1ms jumps.

![Estimated clock drift](image)

**Figure 6. Estimated clock drift in three receivers during 1 hour**

The estimated value $\delta t_{a, SPP}$ does not completely remove the receiver clock error, so the remaining part $\delta t_A$ is estimated in the Kalman filter. We model the remaining part of the clock parameter as a random walk process as recommended by Witchayangkoon (2000). The dynamic model of the remaining receiver clock error $\delta t_A$ is modelled by the following differential equation:

$$\delta t_A = u_{\delta t_A} \tag{2.111}$$

which is written in the general form as in Eq.(2.17), with the following dynamic matrix:

$$F_{\delta t_A} = 0 \tag{2.112}$$

and the coefficient matrix:

$$G_{\delta t_A} = I \tag{2.113}$$

We model the process noise $u_{\delta t_A}$ as white noise with the covariance matrix:

$$Q_{\delta t_A} = E[u_{\delta t_A}(t)u_{\delta t_A}(t)^T] = q_{\delta t_A} I \tag{2.114}$$
The power spectral density of the receiver clocks $q_{\delta r,A}$ is determined empirically and describes how the non deterministic part of the clock error changes in time.

### 2.3.3 Atmospheric delays

The GPS-satellites transmit their signals from an approximate altitude of 20200 km above mean sea level. All signals that reach a GPS-receiver placed on the surface of the Earth have passed through the atmosphere, and they are thus affected by the signal delay. The atmosphere is usually divided into two parts: an ionised and a non-ionised part, according to the presence of charged particles. The ionised part is called the ionosphere and the non-ionised the troposphere.

Subsequent sections give a general description of the atmospheric influence on the propagating signal and how they are modelled in the Kalman filter.

#### 2.3.3.1 Ionosphere

The ionised part of the atmosphere starts at an approximate height of 80 km above sea level. Ionization in the atmosphere is caused by UV and X-radiation from the sun. All gas molecules that are exposed to the radiation are heated and electrons are liberated from them in a process called photo-ionization. Both the ionized molecules and the electrons are charged particles, which influence the propagation of radio waves but it is mainly the electrons that influence the radio. The ionization rate depends on the density of the gas and intensity of the radiation. At low altitudes, where the gas is denser, the charged particles will be recombined rapidly into neutral molecules. Therefore, this part of the atmosphere is almost free from ionized molecules and is called the neutral atmosphere or troposphere. At higher altitudes, where the gas has a lower density, the time before a collision with another particle is increased. Thus, the gas will be full of charged particles, which influence the passing signals. The amount of charged particles will increase with an increase in altitude up to approximately 350 – 400 km; after it starts to decrease. This is mainly because the density of the gas becomes so low at this altitude that even if the ionization of the molecules is more or less total the amount of charged particles is so low that it will not influence a passing signal. The total amount of electrons in the atmosphere is usually measured in units TEC (Total Electron Content), which represents the number of electrons along the signal path from the satellite to the receiver with the size of one square metre.

The radiation depends also on such factors as the geomagnetic latitude, time of day and the presence of ionospheric storms. The magnetic field of the earth influences incoming radiation. The amount of electrons in the atmosphere is higher in the polar areas and at the equator than at latitudes in between them. This implies that the position on earth is an essential factor. The time of day is another factor that influences the radiation. Directly after sunrise the ionization process...
starts in the atmosphere and it continues to increase until approximately 14:00 local time according to Klobuchar (1987), then it fades until the next sunrise. The radiation does not only have a diurnal variation but it also has a long time variation. UV radiation from the sun is changing with a regular pattern which coincides with the number of sunspots. The fluctuation has a period of 11 years and the last maxima were found during 2002.

Besides the factors that directly influence the presence of particles due to photo-ionisation the Sun ejects solar winds, which are streams of high-energy particles. These winds affect the magnetic field of the earth and also indirectly also the ionosphere. Sometimes, massive explosions arise on the surface of the Sun, coronal mass ejections, which cause fluctuations of the geomagnetic field and quick changes in the ionosphere, also called ionospheric storms. These storms are quite difficult to predict but luckily they are of rare occurrence.

The ionosphere does not influence the code and phase observations in the same manner; code observations are advanced and phase observations are delayed. Furthermore, the observations are delayed differently according to their frequency. Leick (2004, pp. 215-219), describes the concept of group and phase propagation through the ionosphere and points out that the ionosphere influences the higher frequencies less than lower. Following Leick’s derivation it is possible to determine that the code advance and phase delay is equal in size but with opposite sign.

2.3.3.1.1 Deterministic delay

There are some methods which can be used to remove or at least reduce the ionospherical influence. Klobuchar (2001) compares the efficiency of some of them. The result is summarised in Table 7.

**Table 2. Summary of the efficiency of different deterministic ionospherical models, Klobuchar (2001)**

<table>
<thead>
<tr>
<th>Efficiency</th>
<th>Type of approach</th>
</tr>
</thead>
<tbody>
<tr>
<td>0%</td>
<td>No model</td>
</tr>
<tr>
<td>50%</td>
<td>Ionospheric Correction Model Algorithm (ICA)</td>
</tr>
<tr>
<td>75%</td>
<td>State of the art ionospheric models like the International Reference Ionosphere (IRI)</td>
</tr>
<tr>
<td>90%</td>
<td>Use Wide Area Augmentation System (WAAS ) or (EGNOS ) ionospheric corrections</td>
</tr>
<tr>
<td>99%</td>
<td>Use dual-frequency receivers</td>
</tr>
</tbody>
</table>
In the first approach the ionospherical delay is ignored totally. The size of an unmodelled ionospheric error can be the size of 20-30 metres. The ICA model is designed by Klobuchar (1987) and it corrects approximately 50% of the ionospheric delay by estimating eight parameters. These parameters are included in the navigation message and are available in real-time. More advanced models like IRI use hundreds of parameters to estimate the ionospheric delay and they manage to model approximately 75% of the delay. A problem with this model according to Klobuchar (2001) is that it does not use any real-time data and they are therefore not appropriate for real-time applications. There are systems, which estimate the ionospheric delay in near real-time, like the Wide Area Augmentation System (WAAS) in the USA and European Geostationary Navigation Overlay Service (EGNOS). In these systems real-time observations from a reference network of permanent GPS receivers, are used to determine the ionosphere delay amongst many other parameters. The corrections are distributed from geostationary satellites, therefore to use these systems it is necessary to have receivers that are prepared to receive the signals and the satellites must be visible from the receiver position. According to Klobuchar (2001) the approach which gives the best result, is to use dual-frequency receivers. Observations from two frequencies can be used to eliminate 99% of the ionospherical delay.

We are using the ICA model, Klobuchar (1987), as the deterministic model for the ionospheric delay. The main reason for this choice is that its parameters are distributed in the navigation message in real-time. So also are the EGNOS parameters but they are only transmitted from the geostationary satellites and the coverage of these satellites is rather limited in the Nordic countries.

The transmitted ICA parameters describe the diurnal curve of the ionosphere which consists of a cosine and a constant part. The cosine part represents the daytime variations and the constant part the night. Both the amplitude and the period of the cosine part vary depending on the receivers’ position on earth. ICA parameters are computed based on the output from an empirical model that describes the world-wide ionospheric behaviour. Normally the parameters are updated each 10 days but if the solar flux value changes (largely during a five day period) then the parameters are updated more frequently.

The input parameters in this algorithm are given here. The complete derivation can be found in Klobuchar (1987) or in ICD-GPS-200C (1999), the interface control document that describes the broadcasted messages. Beware the numerical example given in the former paper as it does not give a correct result. This has been verified by correspondence with the author himself. The general ICA algorithm can be summarised with a function $f$ as:
\[
\Delta I_{ICA} = f(\Phi_A, \lambda_A, El, Az, IP)
\]

Where \(\Delta I_{ICA}\) is the calculated ionospheric delay, \(\Phi_A\) the geodetic latitude and the longitude is denoted by \(\lambda_A\), \(El\) is the elevation angle and \(Az\) the azimuth towards the satellite and IP represents 8 broadcasted ionospheric parameters.

### 2.3.3.1.2 Parameter modelling

To estimate the influence of the ionosphere it is necessary to use observations on at least two frequencies, i.e. it can only be made when dual-frequency observations are available. In single-frequency mode the ionosphere is estimated together with the common error parameters. The estimated parameter \(I^s_A\) in the Kalman filter constitutes the difference between the deterministic model and the true value. The ionosphere is estimated at the reference receivers with one \(I^s_A\) for each satellite/receiver combination. This is necessary since the ionospheric delay is influenced by the azimuth and elevation angle toward the satellite (please refer to the parameters in the Klobuchar model). The ionospheric parameters are not constant in time and therefore they are being modelled as random walk processes with the following dynamic equation:

\[
I^s_A = u_{I,A}
\]

It is not possible to estimate the ionospheric delay at the rover stations since the coordinates are unknown. Instead we use the estimated ionospheres at the reference stations and interpolate the ionospheric influence at the rover stations. The inverse distances between the rover and the reference stations are used as the weights in the interpolation algorithm as described in the following equation with several reference stations (\(A, B, ...\)) and rover station \(C\):

\[
I^s_{i,C}(t) = \frac{d^{-1}_{AC}I^s_A(t) + d^{-1}_{BC}I^s_B(t) + ...}{d^{-1}_{AC} + d^{-1}_{BC} + ...}
\]

The subscript \(i\) indicates that these parameters are interpolated from several stations.

Matrixes \(F_i, T_i, Q_i\) and \(G_i\) for two reference stations and one roving station will be of size \((2 \cdot \text{nsat}) \times (2 \cdot \text{nsat})\), where nsat is the number of satellites \((s = 1, 2, ..., \text{nsat})\), because there are \((2 \cdot \text{nsat})\) parameters to be estimated, namely \(I^s_A\) and \(I^s_B\). (Provided that all of the receivers measure the same set of satellites). The subscript indicates that the matrices belong to the ionospheric delays:

\[
F_i = 0
\]

The \(T_i\) and \(G_i\) matrices are identity matrices and the process noise covariance matrix is given by

\[
Q_{Ai} = \Delta t \cdot \text{diag}(q_i, q_i, \ldots)
\]
where we assume that power spectral density parameters $q_i$ are the same for all reference and rover stations.

### 2.3.3.2 Troposphere

The troposphere is a nondispersive media that has completely different properties than the ionosphere. It is located below the ionosphere, up too height of approximately 50 km above mean sea level and is one of the limiting error sources in GPS-positioning, Mendes and Langley (1994). The troposphere influences all GPS-frequencies in the same way and for that reason the size of its influence on the passing signal is directly related to the travelling distance through the tropospheric layer. Tropospheric refraction is directly related to the amount of water that is mixed in the air. Temperature is one factor that influences the amount of water that can be moisturised in the air. Air with higher temperatures can hold more water vapour than air with lower temperatures. When the maximum of vaporised water is reached at a certain temperature this is called saturated air. Unsaturated air attempts to attain full saturation. When an unsaturated air meets a saturated, they will mix and diffuse to reach full saturation. Furthermore, when the temperature is dropped in a saturated air situation the surplus will condensate until the saturation value is reached at the new temperature. The condensate will be transformed as rain, snow or hail depending on the temperature and fall downwards towards the ground. It will reach the ground if it does not reach another unsaturated air layer. The temperature and also the amount of moisturised air change both temporally and spatially. Consequently the troposphere becomes quite unpredictable. It is well known that the temperature is lower at higher altitudes which results in a higher tropospheric influence at lower altitudes.

To study how a signal is influenced it is possible to divide the troposphere into an infinite amount of thin horizontal layers each with its own refractive index $n$. According to Snell's law, a ray of light that is passing from one refractive index to another will change direction caused by the change of refraction. The path is bends when it passes through the troposphere. Further, the signal velocity will also change when the signal passes through layers with different refractive indexes. The velocity will be reduced in air layers with higher refractive indexes.

The bending of the signal path is usually modelled by a mapping function $m$ which gives the relation between a zenith delay and slant delay at different elevation angles. The equation (2.120) is an example of how a mapping function $m$ is used to map the zenith tropospheric delay (ZTD) to the slant delay $T_{\lambda}^s$ between receiver $A$ and satellite $s$, which is found at an elevation angle $e_{A}^{s}$ above the horizon:

$$T_{\lambda}^s(t) = ZTD(t) \cdot m(e_{A}^{s})$$  \hspace{1cm} (2.120)
To describe the time delay in the troposphere we return to the example with the horizontal thin layers and follow a signal through them along the zenith direction. Since the zenith direction is perpendicular to the horizontal layers, the incoming signal will not be bent, thus the only influence on the signal is caused by the delay. The total zenith tropospheric delay, can be expressed as an integral:

\[ ZTD = \int_{h}^{\infty} n(s) \, ds - \int_{h}^{\infty} \, ds = \int_{h}^{\infty} (n(s) - 1) \, ds = 10^{-6} \int_{h}^{\infty} N(s) \, ds \tag{2.121} \]

where \( s \) is the distance, \( n(s) \) the refraction, both integrated from height \( h \) to \( \infty \). The refractivity \( N(s) \), is normally introduced as a replacement of \( (n(s) - 1) \) in Eq.(2.121), scaled \( 10^6 \) times because the refractive index is very close to unity.

Since the neutral atmosphere consists of both dry air and water vapour, the refractivity can be separated into two parts: one which contains the influence of the dry part \( N_d \) and the other which contain the wet \( N_w \), first suggested by Hopfield (1969):

\[ N = N_d + N_w \tag{2.122} \]

Approximately 90% of the tropospheric refraction is represented by the dry part of the atmosphere and the remaining 10% from the wet part. The problem however is that the wet part is very unpredictable. The size of the dry element is approximately 2.4 meters at main sea level and 0-0.4 metres for the wet part. Thayer (1974) introduced a model to determine the refractivity given as:

\[ N = k_1 \frac{p_d}{T} Z_d^{-1} + k_2 \frac{p_w}{T} Z_w^{-1} + k_3 \frac{p_w}{T^2} Z_w^{-1} \tag{2.123} \]

which is related to partial pressure of dry air \( p_d \) and wet vapour air \( p_w \), with the corresponding compressibility factors \( Z_d^{-1} \) and \( Z_w^{-1} \) of the dry and wet air, the absolute temperature \( T \) given in Kelvin. \( k_1, k_2 \) and \( k_3 \) are empirically determined physical constants. The first two terms in Eq.(2.123) are related to the induced polarisation of air and water vapour molecules respectively. The third term is related to the permanent dipole moment of the water vapour molecules.

Davis et al. (1985) proposed an alternative division of the refractive index. Their idea is to split the refraction into a hydrostatic and a nonhydrostatic part instead of dry and wet constituents. They did this by dividing the dry part in Eq.(2.123) as follows:
where $N_h$ and $N_w$ are introduced to represent the hydrostatic and the nonhydrostatic part. The difference from the previous refraction model is that the first part now has the refractivity of an ideal gas in hydrostatic equilibrium and it is easy to determine with high precision by directly measuring the total pressure $p$ ($p = p_d + p_w$) at the observation site, instead of the partial pressure as in the previous model.

The total slant delay between receiver A and satellite S in Eq. (2.120) can be rewritten for a hydrostatic and a wet part:

$$
T_A^s(t) = T_{h,A}(t) \cdot m_h(e_A^s) + T_{w,A}(t) \cdot m_w(e_A^s)
$$

(2.125)

where $e_A^s$ is elevation angle at the receiver towards the emitting satellite. Subscript h denotes hydrostatic and w wet part of the troposphere.

In the subsequent sections the a priori model for the tropospheric corrections is described. Thereafter, the dynamic parameter models are described and finally the mapping functions.

2.3.3.2.1 Deterministic model

There are several models derived for the tropospheric delay during the years. The most famous models are derived by Hopfield (1969) and Saastamoinen (1972). Both models take metrological data from the surface at the observation site into account. The problem with the metrological observations is that they are not a good representation of the total troposphere since they are influenced by surface layer biases which are introduced by micro-metological effects. There are a priori models for both the hydrostatical and the wet parts of the troposphere that only use a limited amount of metrological observations. A typical example of such a model for the hydrostatical delay is given by Davis et al. (1985):

$$
T_h = 0.0022768 \left( \frac{P_0}{1 - 0.00266 \cos(2 \varphi) - 0.00027 H} \right)
$$

(2.126)

Where $P_0$ is the total atmospheric pressure at the centre of the antenna at the observation site, $\varphi$ the geometric site latitude and $H$ the height above mean sea level in kilometres. The uncertainties of this expression are, according to the authors somewhere, between 0.5 and 20 millimetres per 1000 bar which is slightly better than Saastamoinen's model, with accuracy a on the millimetre level according to Mendes and Langley (1999). They compared this model of the
dry part with radiosonde observations over a year at 50 stations placed all around the world. The denominator in Eq.(2.126) is always close to unity (for reasonable weights) independent of the values of $\varphi$ and $H$ is always close to unity. This means that the main influence on the tropospheric delay is left to the atmospheric pressure, which is changing according to the observation height.

The wet zenith delay has more random nature since it varies according to the amount of water vapour in the air. Both Hopfield (1969) and Saastamoinen (1973), (1973b) and (1973c), derive models for the wet part of the tropospheric delay. Both models where compared with several other models by Mendes and Langley (1999) to find a model that was useful for aircraft applications where metrological observations are rather difficult to perform. They used radiosonde observations to study the root mean squares (r.m.s.) scatter of the models and they found that the size of the scatter were about 5 cm. They presented in the same paper an alternative, slightly improved, model of the zenith wet troposphere:

$$T_w = 0.0122 + 0.00943 p_w$$

(2.127)

This model was originally derived for aircraft applications where accurate observations of the wet partial pressure are rather difficult to determine. Instead they use a data form a standard model from the International Organisation for Standardisation Reference Atmospheres for Aerospace where values can be interpolated according to position and time of year. This model shows the same result as the old models at low latitudes but about 1.6 times better at middle and high levels.

As mentioned before, no metrological observations are carried out at the observation site. To determine the deterministic part of the troposphere we use standard values for the metrological observations with air pressure $1013.25$ hpa, temperature $+20^\circ$C and a relative humidity of 50%.

2.3.3.2.2 Mapping functions

Mapping functions are used to map the tropospheric zenith delay to the actual elevation angle to the satellite. Most of them are derived empirically by adopting equations to radiosonde observations. The mapping functions can be separated into groups by studying their input parameters. Some models use parameters from surface metrology measured on site location, Ifadis (1986), others are based on geometrical parameters, Niell (1996) and there are those, which are based on a combination of both geometrical and atmospheric parameters like Marini (1972). The atmospheric parameters make the model “more difficult to use” while the pressure component is proved to be “weak”, Ifadis (2000). Niell-mapping function is based only on the geometrical parameters at the observation sight and not on any metrological parameters. These properties make it attractive to use, since no metrological observations are required. Several
authors, like Ifadis (2000), Davis et al. (1985) and Mendes and Langley (1994) have shown that this mapping function is useful down to elevation angles of 3 degrees above the horizon.

In Niell mapping function is the same equation Eq.(2.128) which is used to determine both the hydrostatic and wet mapping functions with slightly different parameters:

\[ m(e^S_A) = \frac{1 + \frac{a}{b}}{1 + \frac{1 + c}{\sin(e^S_A) + \frac{b}{\sin(e^S_A) + c}}} + h_{[km]} \left( \frac{1}{\sin(e^S_A)} - \frac{1 + \frac{a_h}{b_h}}{1 + \frac{b_h}{\sin(e^S_A) + \frac{c_h}{\sin(e^S_A) + c_h}}} \right) \]

(2.128)

The procedure to determine the values of \( a, b, c, a_h, b_h \) and \( c_h \) are described by Niell (1996).

2.3.3.2.3 Parameter modelling

The total deterministic tropospheric delay is determined with the equation (2.125) and is subtracted from the observations. The deterministic hydrostatic parameter is determined by using the Davis Eq.(2.126) zenith model for the hydrostatic and Mendes and Langley Eq.(2.127) wet tropospheric delays. These zenith delay values are then mapped to the corresponding slant delay using the Neill mapping function. The remaining unmodelled part of the tropospheric delay is now treated as wet delay, since the wet models are more unpredictable than the hydrostatic. The deterministic model is not perfect and the remaining part is estimated in the Kalman filter at each reference receiver as a random walk procedure.

The dynamic equation that is used for the wet delay \( T_{w,A} \) at reference stations reads:

\[ T_{w,A} = u_{T,A} \]

(2.129)

where the subscript \( w \) indicates that it is a wet delay and \( A \) represents the receiver number. The random part at the rover stations is determined by interpolation at each epoch, just as for the ionosphere.

\[ T_{w,c}^S(t) = \frac{d_{AC}^{-1}T_A^S(t) + d_{BC}^{-1}T_B^S(t) + \ldots}{d_{AC}^{-2} + d_{BC}^{-1} + \ldots} \]

(2.130)

where \( d_{AC} \) corresponds to the distance between the receivers \( A \) and \( C \). The subscript \( i \) represent, as before, an interpolated parameter. Interpolated values are then mapped in to the correct elevation angle with the wet part of the Neill mapping function.

The dynamic matrix \( F_T \) for the zenith troposphere contains only zeros, since it is modelled as a random walk procedure, which means that no dynamic change is introduced during the time between epochs, i.e.:
The state vector contains the estimated parameters for the reference stations; in this case station A and B:

\[ F_T = 0 \]  \hspace{1cm} (2.131)

And the design matrix \( H_T \) and transition matrix \( T_T \) will be identity matrices \( I \). The process noise covariance matrix is given by:

\[ Q_{A,T} = \Delta t \cdot \text{diag}(q_T, q_T, \ldots) \]  \hspace{1cm} (2.133)

It is reasonable to assume that the noise parameters \( q_T \) are the same for all stations and for all satellites.

2.3.4 Receiver antenna and Multipath

Receiver antenna and multipath are both site-dependent effects that influence the GPS-observations. The deterministic part of these site dependent errors is removed with an antenna model that removes the major part of the systematic error that is caused by the used antenna and a multipath model that removes the remaining unmodelled part of the antenna and the influence of the multipath at the observation site.

The multipath model is determined at the observation sites, using calculated residuals from the Kalman filter. Thus, the achieved model is an in-situ calibration model that describes the actual situation at the observation site. Multipath and in-situ calibration of the observation sites are not yet implemented in UGPS so the introduced solution in this section should be seen as a proposal how to proceed.

2.3.4.1 Receiver antenna

All observations measured with a GPS-receiver are related to the phase centres of the receiver antenna. This is not a stable point that coincides with the physical antenna centre, but its influence depends on the frequency and direction of the incoming signal. The influence of the varying phase centre is a systematic error that can be determined in a calibration. The outcome from a calibration is an antenna model that describes the relation between the physical centre of the antenna and the actual phase centre for all incoming directions on the actual frequencies. The physical centre of the antenna is also called the Antenna Reference Point (ARP) and the variations as the phase centre variations (PCV). Ignoring antenna models could result in vertical position errors of up to 10 centimetres according to Rothacher and Schär (1996). This value
shows the importance of using an antenna model. The PCV values are related to the antenna design which means that similar patterns are found for all antennas of the same type.

Three major methods are presently available to determine mean antenna phase centre offsets as well as variation of the phase centre with elevation and azimuth for GPS receiver antennas, (Rothacher 2001):

- Anechoic chamber measurements (laboratory environment)
- Absolute field calibrations
- Relative field calibrations

In the first of these methods the absolute PCV values are determined by using artificial GPS signals transmitted to an antenna placed in an anechoic chamber. The tested antenna is placed on a platform that rotates and tilts with respect to the transmitting source. The phase differences of the generated GPS signal are measured and used to determine the absolute PCV values in all possible directions.

Absolute field calibration is an alternative method to determine the absolute PCV values for a GPS-antenna. The calibration takes place in the field using a high-precision robot that rotates the antenna to be calibrated, while a reference antenna placed a few metres away is fixed. The calibration procedure uses observations from two successive days. During the first day both antennas are fixed whilst on the second day the antenna is mounted on the rotated robot following a special process chart designed specifically so that observations are collected towards all possible antenna azimuths and elevation angles. The final absolute PCV values are determined by comparing the results from the successive days. This method is jointly developed by the Institut für Erdmessung, University of Hannover and the company GEO++, Wübben et al. (1996) and (2000).

The final group of calibration methods is the relative calibration, used at National Geodetic Survey (NGS) USA. Mader and MacKey (1996) and Mader (1999), which describes the calibration procedure in detail; the following represents here follows a brief summary of it. The relative antenna calibration procedure is based on relative observations at two receivers placed only a few metres from each other. At one of the receivers a permanent reference antenna is mounted and at the other the antenna to be calibrated. The calibration procedure is divided into two steps: in the first step constant values of the mean phase centre offset are determined in relation to the ARP. Both antennas are orientated towards magnetic north. This is necessary since the ARP offsets are determined in a local antenna coordinate system (North, East and Up, see Figure 7 where the vector between ARP and the phase centre is represented by dN, dE and
The phase centres for the L1 and L2 channels do not coincide so one set of offset parameters is determined for each frequency.

![Diagram of phase centre](image)

**Figure 7.** The phase centre of an antenna does not always coincide with the physical centre. NGS determines the offsets to the antenna reference point ARP.

In the second step the PCVs’ are determined. These values are necessary since the phase centre is changing with the elevation angles to the satellite. NGS determines PCV values for elevation angels, from 10 to 90 degrees, with a step of 5 degrees, for both satellite frequencies (L1 and L2). The result from the calibrations is summarised in a set of parameters. A template of these parameters is found in Table 3.

The template starts with a header with a description of the tested antenna, how many different antennas that are used to determine the parameters and the date of the parameters. Thereafter follows the parameters for the L1 frequency, with one row with the ARP offsets and two rows with the elevation depending errors. After these rows follows the same set of parameters for the second frequency L2.

**Table 3 Template for the NGS calibration parameters**

<table>
<thead>
<tr>
<th>ANTEONA ID</th>
<th>DESCRIPTION</th>
<th>DATA SOURCE (# OF TESTS)</th>
<th>YR/MO/DY</th>
</tr>
</thead>
<tbody>
<tr>
<td>north</td>
<td>east</td>
<td>up</td>
<td>L1 Offset (mm)</td>
</tr>
<tr>
<td>90</td>
<td>85</td>
<td>80</td>
<td>75</td>
</tr>
<tr>
<td>40</td>
<td>35</td>
<td>30</td>
<td>25</td>
</tr>
<tr>
<td>north</td>
<td>east</td>
<td>up</td>
<td>L2 Offset (mm)</td>
</tr>
<tr>
<td>90</td>
<td>85</td>
<td>80</td>
<td>75</td>
</tr>
<tr>
<td>40</td>
<td>35</td>
<td>30</td>
<td>25</td>
</tr>
</tbody>
</table>

The template values are used to determine distance corrections at different azimuths and elevation angles. Linear interpolation is used for values between the tabulated values. Eq. (2.134) is used to determine the corrections in a certain direction $\Delta P_{CAz,El}$. 

54
\[
APC_{Az,Ei} = \begin{bmatrix}
dN \\
dE \\
dU
\end{bmatrix} + \begin{bmatrix}
\Delta_{PCV}\cos(Az)\sin(Ei) \\
\Delta_{PCV}\sin(Az)\sin(Ei) \\
\cos(Ei)
\end{bmatrix}
\]  

(2.134)

dN, dE and dU are the mean value offset as mentioned before, \(\Delta_{PCV}\) is the interpolated elevation dependent error from the template, Az and El the azimuth and elevation angle of the current observation.

Comparing these calibration methods it is possible to say that the most attractive are the absolute antenna calibration methods. Not only because that they give the absolute PVC values instead of relative values but also because the absolute calibration methods calibrate the antennas down to 0 elevation angle. This is not possible in the relative calibration method because of the multipath and the influence of the atmosphere at low elevation angles.

The general trend within high accuracy GPS positioning is to use absolute calibrated antennas. However, in the absence of absolute calibrated parameters of the used antennas it is possible to use the NGS-antenna corrections in the UGPS software.

### 2.3.4.2 Multipath

Besides the influence of the antenna, there is another error source, multipath, that is influenced by the incoming direction of the signal. Multipath (MPA) arises when a signal bounces before it reaches the antenna. The amount of MPA at a GPS station is related to the properties of the surrounding surfaces and the used GPS-antenna. Thus, the observation sensitivity in respect to MPA can be reduced by the choice of observation site and antenna. An alternative approach to reduce the influence of MPA is to calibrate the observation site. The MPA effect is periodical and it repeat itself in the case of static antenna, as the satellite constellation repeats in one sidereal day (about 23h 56min). If the site around the receiving station does not change from day to day, the signals will follow the same path and thus MPA will remain the same. The size of the MPA influence can be determined with the observation residuals calculated at each epoch as:

\[
MPA_{k+1} = \tilde{I}_{k+1} - h(\bar{X}_{k+1})
\]  

(2.135)

where \(\tilde{I}_{k+1}\) is a vector with the observations that is modified with all the deterministic parameters described in this chapter and \(h(\bar{X}_{k+1})\) is a function which uses the estimated parameter in the actual epoch \((k+1)\) to determine estimated observations in the actual epoch. A typical example of the estimated residuals during three consecutive days is plotted in Figure 8.

The total residuals are separated graphically 0.005 metres and shifted each day by 4 minutes in time to make them comparable.
These residuals can be stored in a look-up table for each satellite and its azimuth and the next day they can be used as “observed” MPA. The values in the table can be updated (re-estimated) every day as a weighted mean from the previous and currently estimated value.

Figure 8. Observation residuals during three consecutive days, each of which is shifted in time so they represent the same epoch. Residuals of day 261 and 262 are shifted +0.005 m and +0.010 m respectively to make the plot readable

The calculated residuals do not only contain the MPA values at the actual station, they will also be influenced by the remaining unmodelled errors in the antenna parameters, the antenna radome (if one is used) and errors caused by the station design, (Granström, 2006). The generated correction values of the site dependent errors are not the true values, in fact, they are more like a set of relative values for the actual total receiver configurations since the estimation of the unknown parameters is performed with observations from several stations that are placed several kilometres from each other. As mention in the introduction to this section, these corrections are not yet implied in the UGPS software and are a subject for further research.

2.3.5 Common errors

All errors that influence the satellite-to-receiver distance, and that is not accounted for by the error sources introduced earlier in this section, are modelled as common errors. Included among the common errors are the satellite positional error and the residual satellite clock error. The size of these errors depends on the quality of the used satellite orbits, (cf. Section 2.4).
To estimate the influence of the ionosphere, observations on two separate frequencies is required. If not, dual-frequency observations are available and the influences of the ionosphere estimated together with the orbital and satellite clock errors in the common error parameter.

All the common parameters are modelled together in a random walk process, which can be described as the following dynamic equation:

$$\dot{\mathbf{u}} = \mathbf{u}_o$$  \hspace{1cm} (2.136)

where $\mathbf{u}_o$ is the forcing function for the common errors. The subscript $o$ indicates common errors. The dynamic matrix for a random walk process is:

$$\mathbf{F}_o = \mathbf{0}$$  \hspace{1cm} (2.137)

the transition and design matrixes will be identity matrices:

$$\mathbf{T}_o = \mathbf{G}_o = \mathbf{I}$$  \hspace{1cm} (2.138)

and the process noise covariance matrix for the common errors is given by:

$$\mathbf{Q}_{o,A} = \Delta t \cdot \text{diag}(q_{o_A}, q_{o_B}, ...)$$  \hspace{1cm} (2.139)

where the power spectral density of the common errors $q_{o_A}$.

### 2.3.6 Observation equations

Inserting the deterministic models into the observation equations Eqs. (2.74) to (2.77) it is possible to obtain a new set of observation equations, where the number of unknown parameters are reduced. The types of estimated parameters dependent upon the receiver is used as a rover or a reference station, e.g. where the station coordinates are known at the reference stations and thus not estimated. This implies that the number of unknown parameters at the reference stations is less than at rover stations and this is why it is suitable to estimate the atmospheric influence at the reference stations. In the following two subsections the updated observation equations are presented.

#### 2.3.6.1 At the reference stations

The code observation equations for the reference station $A$ are given by:

$$\delta p_{A,1}^C(t_A) = (\rho_A^C + c)\delta t_A + m_{w,A}^S T_{w,A}(t_A) + \delta \alpha_A^S(t_A) + \delta \theta_A^S(t_A) + \epsilon_{p1}$$  \hspace{1cm} (2.140)

and

$$\delta p_{A,2}^C(t_A) = (\rho_A^C + c)\delta t_A + m_{w,A}^S T_{w,A}(t_A) + \alpha t_A^S(t_A) + \delta \theta_A^S(t_A) + \epsilon_{p2}$$  \hspace{1cm} (2.141)
where $\delta P_{A,1}$ and $\delta P_{A,2}(t_A)$ on the left side in these equations represent the observation equations after all deterministic parameters are removed, and on the right side are the remaining parameters to be estimated in the adjustment procedure. The unknown stochastic parameters in these equations are:

- $\delta t_A$: The receiver clock
- $I_A^S$: The Ionospheric code delay
- $T_{w,A}$: The wet part of the tropospheric delay
- $\delta O_A^S$: Common errors

and the other non stochastic parameters:

- $\alpha_I$: The constant that describe the relationship between the frequencies, (introduced in Section 2.2.2.1.3)
- $\dot{\rho}_A^S$: Velocity in geometric range at the true time of the receiver
- $c$: The speed of light in vacuum
- $m_{w,A}^S$: Mapping function for the wet part of the tropospheric delay
- $t_A$: The nominal time in the when the signal reaches the antenna
- $\varepsilon_{p1}$ and $\varepsilon_{p2}$: The pseudo range measurement noise.

The corrected observations on the left sides of the equal sign are given by:

$$\delta P_{A,1}^S(t_A) = P_{A,1}^S(t_A) - \rho_A^S(t_A) - (\rho_A^S + c)\delta t_{A,SPPP} + c s_{c,A} + c(\delta t^S - T_{GD}^S) - m_{d,A}^ST_{n,A}^S(t_A) - I_{d,A}^S(t_A) - \delta A_{A,P1}^S(t_A) - \delta M_{A,P1}^S(t_A) \quad (2.142)$$

and

$$\delta P_{A,2}^S(t_A) = P_{A,2}^S(t_A) - \rho_A^S(t_A) - (\rho_A^S + c)\delta t_{A,SPPP} + c s_{c,A} + c(\delta t^S - \alpha_I T_{GD}^S) - m_{d,A}^ST_{n,A}^S(t_A) - I_{d,A}^S(t_A) - \delta A_{A,P2}^S(t_A) - \delta A_{P2}^S(t_A) \quad (2.143)$$

where the deterministic parameters are:

- $P_{A,1}^S$ and $P_{A,2}^S$: The raw unmodified observations
- $\rho_A^S$: The distance between the station and satellite calculated with the known satellite and receiver coordinates, the estimated clock parameter
- $\delta t_{A,SPPP}$: The common parameter that estimates the deterministic part of the drift and bias in the receiver clocks

58
The integer that keeps record on the number of detected 1ms clock corrections

\( \delta t^S \) The satellite clock correction

\( T_{GD}^S \) The group delay that is distributed in the broadcast ephemerides

\( I_{d,A}^S \) The deterministic part of the ionosphere

\( T_{h,A} \) The deterministic hydrostatic part of the troposphere

\( m_{h,A}^S \) Dry mapping function for the hydrostatic part of the troposphere

\( \delta A_{A,p1}^S \) Antenna offset at the receiver

\( \delta A_{A,s} \) Antenna offset at the satellite

\( \delta M_{A,p1}^S \) Influence of the multipath

The hardware delay parameters have been removed from these equations since, its influence is assumed to be covered by the satellite clock parameter and the corresponding delay in the receivers is assumed to be removed in the deterministic part of the receiver clock delay, determined in the single point positioning algorithm (see Section 2.3.2).

The corresponding phase observation equations at the reference receivers become:

\[
\delta \Phi_{A,1}^S(t_A) = (\dot{\rho}_{A}^S + c) \delta t_A + m_{w,A}^S T_{w,A}(t_A) - I_{d,A}^S(t_A) + \delta O_{A}^S(t_A) + \lambda_{L1} N_{A,L1}^S + \epsilon_{L1} \tag{2.144}
\]

and

\[
\delta \Phi_{A,2}^S(t_A) = (\dot{\rho}_{A}^S + c) \delta t_A + m_{w,A}^S T_{w,A}(t_A) - \alpha t_{A}^S(t_A) + \delta O_{A}^S(t_A) + \lambda_{L2} N_{A,L2}^S + \epsilon_{L2} \tag{2.145}
\]

where the left sides are:

\[
\delta \Phi_{A,1}^S(t_A) = \Phi_{A,1}^S(t_A) - \rho_{A}^S(t_A) - (\dot{\rho}_{A}^S + c) \delta t_{A}^{SPP} + c s_{c,A} + c \delta t^S - m_{d,A}^S T_{d,A}(t_A) + I_{d,A}^S(t_A) - \delta A_{A,L1}^S(t_A) - \delta A_{L1}^S(t_A) - \delta M_{A,L1}^S(t_A) \tag{2.146}
\]

and

\[
\delta \Phi_{A,2}^S(t_A) = \Phi_{A,2}^S(t_A) - \rho_{A}^S(t_A) - (\dot{\rho}_{A}^S + c) \delta t_{A}^{SPP} + c s_{c,A} + c \delta t^S - m_{d,A}^S T_{d,A}(t_A) + \alpha I_{d,A}^S(t_A) - \delta A_{A,L2}^S(t_A) - \delta A_{L2}^S(t_A) - \delta M_{A,L2}^S(t_A) \tag{2.147}
\]

The unknown parameters are the same in these equations as in the code observation equations, except for the ambiguity parameters \( N_{A,L1}^S \) and \( N_{A,L2}^S \). \( \lambda_{L1} \) and \( \lambda_{L2} \) represent the wavelengths on L1 and L2 frequencies.
2.3.6.2 At the rover stations

Following the same procedure in removing the deterministic parameters in the observation equations, at a rover receiver \( C \) the observation equations becomes:

\[
\delta P_{\text{C,1}}^S(t_c) = a_x \Delta X + a_y \Delta Y + a_z \Delta Z + (\rho_0^S + c) \delta t_c + m_w^S T_{w,c}(t_c) + I_c^S(t_c) + \delta O_c^S(t_c) + \epsilon_{p1} \tag{2.148}
\]

and

\[
\delta P_{\text{C,2}}^S(t_c) = a_x \Delta X + a_y \Delta Y + a_z \Delta Z + (\rho_0^S + c) \delta t_c + m_w^S T_{w,c}(t_c) + \alpha_t I_c^S(t_c) + \delta O_c^S(t_c) + \epsilon_{p2} \tag{2.149}
\]

The rover observation equations include more unknown parameters than the corresponding at the reference stations. Besides all the parameters that are estimated at the reference stations is also a set of coordinate corrections (\( \Delta X, \Delta Y, \Delta Z \)) estimated. These are used to update the approximate coordinates that are given as inputs to the filter (cf. Section 2.4.1). The approximate coordinates are used to determine the distance between receiver and satellite \( \rho_{0,c}^S \) that is used on the left side of the observation equations, see Eqs (2.150) and (2.151). If these coordinates are not initially given to the Kalman filter we calculate them with the same SPP algorithm that is used to estimate the clock bias. On the other hand, when the filter is running, the approximate coordinates are taken directly in the predicted state vector of the Kalman filter.

The influences from the tropospheric and Ionospheric delays are not estimated directly at the rover stations. The estimated values are not explicitly determined, e.g. when two receivers are used, a reference and a rover, we have to assume that the tropospheric influence is the same at both stations when the deterministic part of the atmospheric influence is removed since we do not have any additional information about the troposphere. This implies that the estimated troposphere at the rover receiver is exactly the same as at the reference receiver (\( T_{w,a} = T_{w,c} \)). This approach is normally used in RTK applications based on double differences where the assumption that the double differenced atmospheric influences is so small that they can be ignored. When more than one reference receiver is used is weighted values of the atmospheric influence estimated at the rover stations according to the weighting procedures introduced in Section 2.3.3.

The left side of the rover observation equations are:

\[
\delta P_{\text{C,1}}^S(t_c) = P_{\text{C,1}}^S(t_c) - \rho_{0,c}^S(t_c) - (\rho_0^S + c) \delta t_c + c \delta t^S - \delta A_{\text{C,P1}}^S(t_c)
\]

\[
+ m_d^S T_{d,c}(t_c) - \delta A_{\text{C,P1}}^S(t_c) - \delta t^S - T_{d,c}(t_c) - I_d^S(t_c) - \delta A_{\text{C,P1}}^S(t_c)
\]

\[
- \delta A_{\text{P1}}^S(t_c) - \delta M_{\text{P1}}^S(t_c) \tag{2.150}
\]

and
\[ \delta p_{\text{C},z}(t_c) = p_{\text{C},z}(t_c) - \rho_{\text{C},0}(t_c) - (\dot{\rho}_c^s + c) \delta t_{\text{C},\text{SPP}} + c s_{c,c} \\
\quad \quad + c(\delta t^f - \alpha_{\text{TGD}}^s) - m_{d,c}^s T_{h,c}(t_c) - \alpha_d^s I_{d,c}^s(t_c) - \delta A_{c,p2}^s(t_c) \\
\quad \quad - \delta A_{d2}^s(t_c) - \delta M_{c,p2}^s(t_c) \tag{2.151} \]

The deterministic parameters are the same as at the reference stations. The corresponding phase observation equations on the rover stations are given as:

\[ \delta \Phi_{\text{C},1}^s(t_c) = (\dot{\rho}_c^s + c) \delta t_c + m_{w,c}^s T_{w,c}(t_c) - I_{n}^s(t_c) + \delta O_{c}^s(t_c) + \lambda_{L1}^s N_{c,L1}^s + \epsilon_{L1} \tag{2.152} \]

and

\[ \delta \Phi_{\text{C},2}^s(t_c) = (\dot{\rho}_c^s + c) \delta t_c + m_{w,c}^s T_{w,c}(t_c) - \alpha_d^s I_{d,c}^s(t_c) + \delta O_{c}^s(t_c) + \lambda_{L2}^s N_{c,L2}^s + \epsilon_{L2} \tag{2.153} \]

where

\[ \delta \Phi_{\text{C},1}^s(t_c) = \Phi_{\text{C},1}^s(t_c) - \rho_{\text{C},0}^s(t_c) - (\dot{\rho}_c^s + c) \delta t_{\text{C},\text{SPP}} + c s_{c,c} + C \delta t^s \\
\quad \quad - m_{d,c}^s T_{h,c}(t_c) + I_{d,c}^s(t_c) - \delta A_{c,L1}^s(t_c) - \delta A_{l1}^s(t_c) - \delta M_{c,L1}^s(t_c) \tag{2.154} \]

and

\[ \delta \Phi_{\text{C},2}^s(t_c) = \Phi_{\text{C},2}^s(t_c) - \rho_{\text{C},0}^s(t_c) - (\dot{\rho}_c^s + c) \delta t_{\text{C},\text{SPP}} + c s_{c,c} + C \delta t^s \\
\quad \quad - m_{d,c}^s T_{h,c}(t_c) + \alpha_d^s I_{d,c}^s(t_c) - \delta A_{c,L2}^s(t_c) - \delta A_{l2}^s(t_c) \\
\quad \quad - \delta M_{c,L2}^s(t_c) \tag{2.155} \]

### 2.4 Software implementation

The undifferenced approach seems to have some attractive properties compared with the differenced approach, especially in real-time multi-station situations. To evaluate this method is software package developed. The first version of this software was developed in Matlab© for evaluation of the algorithms,(Andersson 2006). The software is running under the working name UGPS since it uses undifferenced GPS observations. The main problems with the Matlab implementation were the low computational speed and that no more than one reference station could be used when real observations are used as input data. To improve the computational speed, UGPS was converted into C++ code and compiled into a windows application. All matrix calculations are made with the use of the CVM class library ([www.cvmlib.com](http://www.cvmlib.com)). This library uses the Basic Linear Algebra Subprograms (BLAS) and Linear Algebra PACKage (LAPACK) Fortran libraries in order to achieve the best numerical performance possible, (Nikolaev 2007). During the conversion into C++ also algorithms were developed, which allow the software to use several reference receivers at the same time.
The displacement and alarm part of the system is developed in Matlab as a separate module for evaluation. It uses the output parameters from UGPS as input data. More details about the used automatic displacement monitoring algorithms are given in Section 2.5.

So far, the routines for reading observations from GPS-receivers are not implemented. Since this project is limited in time, the focus is placed on the performance of the system and its performance in detecting displacements. Consequently, UGPS cannot run in real-time, but all its algorithms are developed and uses all input data just as it would.

The working procedure in real-time software is slightly different than in the corresponding post processing software. In a post processing software is it possible to improve the estimated parameters since the complete set of observations from all receivers available before the calculations are started. The accessibility makes it possible to screen the observations to find errors and outliers within them. Furthermore, the broadcasted ephemerides can also be checked for inconsistencies before use or even better precise ephemerides could be used e.g. IGS precise ephemerides. These ephemerides are available with a delay of 2 weeks. Further improvement could be found by using more accurate models of the troposphere and ionosphere which also are available after a few weeks, (IGS-homepage http://igscb.jpl.nasa.gov).

A real-time software can only use the observations up to the current epoch and satellite orbits that are accessible in real-time. This includes the actual observations and the broadcasted ephemerides. If the software is connected to internet, it is possible to use predicted part of IGS-ultra rapid orbits.

An overview of the implemented algorithms in the UGPS-software is presented in Figure 9. The algorithm starts at the top of the figure with an initialisation procedure, followed by the Kalman filter algorithm, which is repeated each by epoch until all epochs are processed. The estimated state vector is stored inside the Kalman filter and extracted to ASCII-files when the calculations are finished. These data is then used in displacement detection algorithms.

The initialisation procedure contains two steps: in the first step the initialisation file that contains start values of all parameters in the filter and the filenames of all related files is read. In the second step of the initialisation are the state vector $\hat{x}_0$ and the covariance matrix $Q_{x_0}$ of the Kalman-filter filled in with the given start values initialised with the given start values and the first common epoch in the observation files is found. UGPS reads observations and ephemerides in RINEX format, Gurtner (2001), and antenna models are given in NGS-antenna format, see Table 3. When the initialisation is ready, the post processing program is ready to start.

The Kalman filter loop includes several separate steps. In the first step the observations at all receivers are read from the observation files. The observations are stored in a class structure
with the satellite PRN number as identifiers. In the subsequent step the satellite coordinates are determined for the actual epoch for all the used satellites. These are calculated with a set of "standard orbits" (Horemuž and Andersson 2006). A new set of standard orbits is calculated each time new ephemerides data is available, e.g. the broadcasted ephemerides are updated with an interval of maximum 2 hours. This approach is presented in Section 2.4.2.1.

After the initialisation follow a series of pre-processing steps that starts with prediction of the state vector and the covariance matrix into the current epoch. All deterministic parameters are determined in the succeeding step and used to adjust the observation equations according to the equations given in Section 2.3. The deterministic part of the receiver clock is determined with a SPP-algorithm. The output from this algorithm is not only the estimated clock parameters, but also initial values for the ionosphere and ambiguities. The pre-processing step is continued with calculation of the observation weights and finally the observations are checked for cycle slips. The used observation weighting and cycle-slip detection algorithms are discussed further in Section 2.4.2.2 and 2.4.2.3.

When the pre-processing of the observations is finished we have a set of observations that are corrected with the deterministic parameters, and all new parameters have approximate values in the state vector. Now it is time to fill in the observation vector \( \mathbf{L} \) as well as the design matrix \( \mathbf{H} \) and the corresponding covariance matrix \( \mathbf{R} \). The size of the state vector and its corresponding covariance matrix in the Kalman filter is not constant; it has to be adjusted in size so that it agrees with the actual observations in the observation vector. States are added when new satellites arrive and are removed when they disappear. When all matrices are filled the process continues with the Kalman filter update, where the predicted parameters are blended together with the actual observations in the current epoch. The last step of the filter loop is ambiguity fixing, which is entered, when the filter has passed a certain number of epochs. The ambiguity fixing is based on the lambda method developed at the University of Delft and is described in Section 2.4.5. At the end of a loop in the Kalman filter the updated unknown parameters in the state vector are presented with their standard errors. Each of the steps in the Kalman filter is studied in detail in the following subsections.

The output from the UGPS software is an ASCII file with all the deterministic and estimated parameters together with their standard deviations. These values are used to study the performance of the filter and to evaluate different displacement monitoring algorithms. More about these algorithms is found in Section 2.5.
Figure 9. Overview of the processing scheme in the UGPS-software
2.4.1 Initialisation

The process in UGPS is controlled by the use of an initialisation file. This file contains information about process settings, start values of the unknown parameters, input files and output parameters. The following parameters are used to control the process:

- **Minimum elevation angle mask**, is used to remove all observations that are made to satellites under the minimum value
- **Minimal signal strength**, controls the minimal allowed signal strength in the observations
- It is possible to choose between the *Observation weighting methods* given in Section 2.4.2.2
- Single or Dual-frequency observations, L2 observations can be removed from the observations to perform calculations in single-frequency mode
- The *troposphere estimation* can be switched on and off and it is possible to choose different deterministic models
- **Minimum number of epochs** before the ambiguity fixing procedure. The estimated parameters in the Kalman filter will improve over time when more observations are used in the estimation. By setting a minimum number of epochs before the ambiguities are fixed it is possible to ensure that the parameters have stabilised before the ambiguities are fixed.
- **Instrument heights**, is used if the antenna reference point is not the actual fixed marker to be determined
- **Input files**, as observations, antennas and orbits are also specified here.

Start values are needed for all the estimated parameters to start up a Kalman filter. The needed start values contain initial values of the unknown parameters together with standard deviation that describes the precision of the parameter and a Power Spectral Density (PSD) value that controls the amount the value is allowed to change over time. The used start values are found empirically and are presented in Table 4.

2.4.2 Pre-processing

The pre-processing consists of five steps; the Kalman filter prediction, determination of all deterministic parameters, initialisation of new parameters, and calculation of observation weights and finally a cycle slip detection step.

In Section 2.1.3.1 the prediction step of the Kalman filter is described and in Section 2.3 are most of the deterministic parameters introduced, and therefore these parameters not studied further here. Instead the focus is placed on parameters that are not earlier introduced, as the satellite positioning and observation weighting algorithms.
Initialisation of new parameters follows the same procedure as when the filter is started. Thus, when a new satellite occurs all related parameters are introduced in the state vector, with initial values that are calculated in the same single point positioning algorithm as used to compute the deterministic part of the receiver clocks; see Section 2.3.2. Instead the final step of the pre-processing procedure, the cycle slip detection studied in detail. It will be shown that it is important to find all cycle slips since the algorithm is very sensitive to undetected cycle slips.

**Table 4. Initial values used in the Kalman filter**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Initial values</th>
<th>Standard deviation</th>
<th>PSD</th>
</tr>
</thead>
<tbody>
<tr>
<td>Position</td>
<td>From header*</td>
<td>5 m</td>
<td>-</td>
</tr>
<tr>
<td>Velocity</td>
<td>0</td>
<td>0.001 m/s</td>
<td>0.001</td>
</tr>
<tr>
<td>Receiver clock error</td>
<td>0</td>
<td>100 m</td>
<td>0.5</td>
</tr>
<tr>
<td>Troposchere</td>
<td>0</td>
<td>0.2 m</td>
<td>0.005</td>
</tr>
<tr>
<td>Ionosphere</td>
<td>0</td>
<td>30m</td>
<td>0.0001</td>
</tr>
<tr>
<td>Ambiguities</td>
<td>From clock estimation</td>
<td>400 cycles</td>
<td>-</td>
</tr>
<tr>
<td>Common errors</td>
<td>0</td>
<td>50</td>
<td>0.5</td>
</tr>
</tbody>
</table>

* Initial coordinates are found in the header of the observation file in RINEX format. Alternatively, the navigation solution obtained from the GPS-receivers can be used.

**2.4.2.1 Satellite positions**

Satellite positions in real-time procedures can be calculated either with use of the broadcasted ephemerides or, if internet available, with predicted ultra rapid orbits generated supplied by IGS.

The broadcast ephemerides are represented by a set of Kepler orbit parameters with correction terms and the precise orbits are given by satellite coordinates with a sample interval of 15 minutes. The broadcast ephemerides are updated approximately every 2 hours, and the predicted precise ephemerides are updated each 24 hours. The final precise ephemerides are not available in real-time, they are determined in a post processing procedure which uses observations from several days to calculate the orbits. The accuracy in the broadcast ephemerides is about 3 metres and 0.05 metre for the final post-processed coordinates.

When broadcasted parameters are used to calculate the satellite coordinates, the standard algorithms are used, as described in document ICD-GPS-200C. In the case when precise ephemerides are used then the satellite coordinates are determined by interpolation in the tabular coordinates.
Instead of using two different algorithms to determine the satellite coordinates, we have developed a general algorithm that is independent of the ephemerides input type implying that, both precise and broadcasted ephemerides can be used. The basic idea with the algorithm is to interpolate satellite coordinates, for a given epoch, using a set of polynomial coefficients that describes the satellite orbits. The polynomial coefficients form what we call the standard orbits. These are determined by fitting a polynomial of a certain degree, in a least squares sense, to satellite coordinates that are given in fixed intervals. The polynomial is given by:

\[ p(t) = a_1 t^n + a_2 t^{n-1} + \cdots + a_n t + a_{n+1} \]  

in which \( t \) is the time in seconds from the beginning of the time interval and \( a_n \) is the polynomial coefficients of order \( n \). The time interval we use is 15 minutes, just as in precise orbits, and we call it tabular orbits. Broadcasted ephemerides are transformed into tabular orbits simply by calculating the satellite coordinates and clock corrections by the standard algorithm at the wanted time interval. The general real-time satellite calculation algorithm is given in Figure 10.

![Figure 10. General flowchart for calculating satellite positions](image)

When the algorithm is started the first step is to check if precise ephemerides (PE) are available. If the answer is yes, then the algorithms directly starts to calculate the standard orbits and if it is no then the broadcasted ephemerides are used instead, where an extra step is needed to create the standard orbits.

If a polynomial is fit to equidistant data, oscillations can occur at the beginning and the end of the interval. This is a known problem in numerical analyses where these oscillations are called Runge’s Phenomenon (RP). To avoid RP the first and last part of the fit interval are removed and
the remaining part is named as validity interval. In Figure 11 validity interval and the fit interval are shown for 4 hour fit interval and 2 hour validity interval.

![Validity interval and fit interval](image)

**Figure 11. Fit and validity intervals**

A typical example of RP can be seen in Figure 12, where the difference between interpolated coordinates is compared with coordinates that are calculated with the direct formulas. The oscillations are obvious at the start and end of the 3 hour interval.

![Runge's phenomenon](image)

**Figure 12. Runge's phenomenon on a 3 hour fit interval**

A new set of standard orbits are calculated as soon as time has reached the end of the validity interval or if new ephemerides parameters are found.

The performance of this algorithm is studied in Horemuž and Andersson(2006) and they found that optimal performance is obtained when a 4 hour fit interval is used together with a 2 hour validity interval and a maximum polynomial order is used, which in this case is 16.
2.4.2.2 Observation weighting

In positioning with undifferenced observations, based on a Kalman filter, correct stochastic and functional models are needed both for the time dynamic process and for the observations. The functional and stochastic models for the dynamic process in the Kalman filter are described in Section 2.4.1. Here, we study different weighting schemes for the observations. The observations are assumed to be uncorrelated, thus the variance for each observation is placed on the diagonal in the covariance matrix $R$. Typical factors that influence the GPS observation noise level is the receiver dependent noise, diffraction and multipath. Since these parameters vary during time, new weights are needed at each epoch.

The receiver dependent noise is related to the accuracy in the correlation procedure, which is performed by the GPS-receiver. In Section 2.2.1.3 a rule of thumb was introduced that said that the noise level of the signal is about 1% of the signal wavelength. This implies that a phase observation would have the noise level of approximately 2 millimetres, the wavelength is about 0.2 metres, and about 3 metres for the code observations where the wavelength is 300 metres.

Table 5. Comparison: Diffraction and Multipath effects (Wanninger et al. 1999)

<table>
<thead>
<tr>
<th>Common features</th>
<th>Diffraction Effects</th>
<th>Multipath Effects</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Depends on local environment of receiving antenna</td>
<td>Superposition of direct and reflected (indirect) signals</td>
</tr>
<tr>
<td></td>
<td>Repeats with identical satellite constellations and unchanged environment</td>
<td>Frequency dependent: detected in the geometry free combination</td>
</tr>
<tr>
<td></td>
<td>No mitigation by relative positioning</td>
<td>Maximum error $\lambda_s/4$</td>
</tr>
<tr>
<td></td>
<td>The effects on coordinate estimation is reduced with increase of observation time (static observations)</td>
<td>Fluctuation in the signal strength</td>
</tr>
<tr>
<td></td>
<td>In kinematic mode rapid changes of diffraction effects: mitigation by filtering</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Differences</th>
<th>No line-of-sight, only diffracted signal are received</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Independent of signal frequency: geometry free linear combination are not affected</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Maximum error; in the order of decimetres</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Signal strength reduced</td>
<td></td>
</tr>
</tbody>
</table>

In addition to the system dependent noise, there are other noise sources like diffraction that occurs when something blocks the true signal path between a satellite and a receiver but the signal still arrives to, and is recorded by the receiver. Multipath occurs when the signal bounces before it reaches the antenna. There are some similarities and some differences between diffraction and multipath that are useful to have in mind. Wanninger et al. (1999) summarises these in Table 5.
2.4.2.2.1 Weighting methods

The simplest method that can be used in observation weighting is to apply equal weights to all observations of the same type. A problem with this method is that it does not treat the fact that the noise level is increased at lower elevation angles. The normal method to avoid this is to remove all observations to satellites that are visible below a certain elevation angle. The drawback of this approach is that the number of satellites becomes less, which results in a poorer satellite geometry that also influences the positioning accuracy negatively.

To avoid this, more sophisticated weighting methods have been developed. These methods mainly use two different parameters: elevation angle and signal-to-noise ratio (SNR). All the factors (diffraction, multipath and receiver dependent noise) are influenced by the elevation angle, (Collins and Langley1999). This makes the elevation angle a good parameter, which can be used for observation weighting.

The other parameter, SNR, can be derived directly from each observation. The SNR is usually represented by the carrier-to-noise ratio (C/No) which is a normalised SNR value and represents the ratio of the power level of the signal carrier to the noise power in a one Hertz bandwidth. C/No values have a specific place in observation files given in RINEX-format.

A weighting approach that uses the elevation angel towards the satellite was introduced by Wieser and Gaggl (2005) as:

\[
\sigma^2_E = \frac{a_0}{\sin^2 E}
\] (2.157)

where \(E\) is the elevation angle and \(a_0\) a coefficient that is estimated empirically, see Wieser and Gaggl (2005). When \(a_0\) is determined it is easy to determine the a-priori variances \(\sigma^2_E\) for the variance-covariance matrix \(R\) with use of the elevation angle \(E\). The weight matrix is then computed as the inverse of \(R\).

Hartinger and Brunner (1999) and Brunner et al. (2000) use a method that they call the sigma-e, which is based on a similar equation as the elevation based weighting method but instead of the elevation angle they use the C/No values.

\[
\sigma^2_E = a_0 + a_1 \cdot 10\left(\frac{C/No}{10}\right)
\] (2.158)

Again, a preliminary set of observations is used to estimate the coefficients \(a_0\) and \(a_1\). This weighting model shows a very good performance, and it reduces the residuals of the baseline component approximately 40 % according to their result.

The research group within Engineering surveying and Metrology at the University of Graz have developed several improvements to the sigma-e model. Sigma-\(\Delta\) were introduced Brunner et al.
(2000), which is also based on the C/No values and sigma-F by Wieser and Brunner (2000), which is based on fuzzy algebra. The authors show that both these methods improve the result compared with the sigma-e method.

A comparative study of different quality indicators used in observation weighting is done by Satirapod and Wang (2000). They concluded that the weighting procedure based on C/No gives the best result. They also studied the C/No values from several receivers connected to the same antenna and concluded that the C/No values is receiver dependent but with similar patterns. This implies that one set of weighting coefficients are necessary for each receiver to determine the correct weights.

All the presented weighing methods are implemented in the software UGPS. A test to analyse the performance of different weighting methods in the actual positioning algorithms based on undifferenced observations have been made in (Andersson, 2006 pp.102-103), here reprinted in Table 6.

Table 6. Different weighting models in the kinematic calculations of baselines. Model 1 is equal weighting, model 2 elevation dependent weighting and model 3 is C/No based weighting

<table>
<thead>
<tr>
<th>Baseline</th>
<th>Weight Type</th>
<th>Mean Value (mm)</th>
<th>STD (mm)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Model</td>
<td>N</td>
<td>E</td>
</tr>
<tr>
<td>HIS-ROV1</td>
<td>1 SF</td>
<td>0.0</td>
<td>0.3</td>
</tr>
<tr>
<td>HIS-ROV1</td>
<td>2 SF</td>
<td>-0.1</td>
<td>0.4</td>
</tr>
<tr>
<td>HIS-ROV1</td>
<td>3 SF</td>
<td>0.2</td>
<td>0.3</td>
</tr>
<tr>
<td>BAG-ROV1</td>
<td>1 SF</td>
<td>0.0</td>
<td>0.6</td>
</tr>
<tr>
<td>BAG-ROV1</td>
<td>2 SF</td>
<td>1.1</td>
<td>1.0</td>
</tr>
<tr>
<td>BAG-ROV1</td>
<td>3 SF</td>
<td>1.6</td>
<td>-0.8</td>
</tr>
</tbody>
</table>

Two baselines are calculated in single-frequency (SF) mode, one short (HIS-ROV1, 456m) and one long (BAG-ROV1, 1887m). In the second column is the actual weight model presented, equal weighting is applied in 1, the elevation dependent approach in 2 and finally the C/No model in 3. In the mean value column the difference between the known position, determined with static calculations, and the mean value of all the kinematic positions is presented. Finally, in the STD columns the standard deviations in the estimated coordinates are presented. The outcome from this test confirms the result from Satirapod and Wang (2000), that the (C/No) weighting method performs better than the other above presented weighting methods.

2.4.2.3 Cycle slip detection

When a GPS receiver is measuring in phase observations, it records the fractional phase difference between the incoming phase and the generated phase observation and counts the
phase shifts when the fractional phase switches from $2\pi$ to 0. The counting is performed continuously, from the moment the start of observation. If, by some reason, the signal between the receiver and satellite is interrupted, the integer counter has to start over again, which causes a jump in the continuous count of phase shifts. These jumps are called cycle slips and they need to be detected to determine accurate positions. Sjöberg (2005, p.74) summarizes the reasons that cycle slips occur:

- An obstacle disturbance of the ray path
- Too low signal-to-noise ratio (e.g. as a result of multi-path, ionosphere bias, large receiver accelerations, etc.)
- Failure of the satellite oscillator

All high precision positioning with GPS based on phase observations are sensitive to cycle slips, e.g. Figure 11 shows how an undetected cycle slip of 1 wavelength (L1) influences the estimated coordinates at a rover station in the undifferenced approach.

![Influence of undetected cycle slip](image)

**Figure 13. Influence of an undetected cycle slip**

The general procedure to detect a cycle slip is to study time series that are calculated with some combination of observation that are less sensitive to disturbances than the raw observations. In the following sections some of methods to identify cycle slips presented together with a method that uses the predicted residuals of the Kalman filter to detect the cycle slips.

When a cycle slip is detected in the pre-processing step the estimated ambiguity in the state vector $\hat{\mathbf{x}}$ is removed together with the corresponding values in the covariance matrix $\mathbf{Q}_x$. Thereafter, new values are determined and inserted into $\hat{\mathbf{x}}$ and $\mathbf{Q}_x$, following the same procedure as in the initialisation step.
2.4.2.3.1 Single-frequency phase / code combinations

A combination of code and phase observations can be used to detect cycle slips in the case of single frequency observations. The difference between the code and phase observation \( R_{A,LI}^A \) results in a value that is strongly related to the ionosphere and the ambiguities. This can be seen on the right side of Eq (2.159):

\[
R_{A,LI}^A = p_A^S - \Phi_A^S = 2f_A^S - \lambda N_A^S + cT_{GD} + \delta_R + \varepsilon_{A,R}^S
\]  

where the new parameters are \( \delta_R \), that contains the multipath, and \( \varepsilon_{A,R}^S \) being the measurement noise on the phase and code observations. The difference between the values of \( R_{A,LI}^A \) at epoch \( t \) and epoch \( t_{k+1} \) will become very small if the time between epochs is short and if no cycle slips occur during the time between the epochs. This conclusion can be drawn with the knowledge that the ionosphere and the multipath usually change very slowly in time and that the group delay is constant. The problem with this method is the noise level of \( R_{A,LI}^A \), which is in the range of ±5 cycles, (Hoffmann-Wellenhof et al. 2001, p.208), mainly caused by the code observations.

2.4.2.3.2 Dual-frequency phase combinations

Cycle slip detection can be performed by differencing the phase observations of the two frequencies L1 and L2:

\[
\Phi_A^S = \Phi_{A,L2}^S - \Phi_{A,L1}^S = \lambda_{L2} N_{A,L2}^S - \lambda_{L1} N_{A,L1}^S + \left( 1 - \frac{f_1^2}{f_2^2} \right) I_A^S + \delta_\Phi + \varepsilon_{A,\phi}^S
\]  

where \( \delta_\Phi \) and \( \varepsilon_{A,\phi}^S \) are the multipath and the observation noise for the phase combination. The precision in the phase observations is higher than for the code ones so this method is a better method than the single-frequency combination presented in the previous section. Based on the rule of thumb about the noise level introduced in Section 2.2.1, it is possible to determine cycle slips jumps up to ± 4 cycles when comparing the differences calculated at two subsequent epochs, (Hoffmann-Wellenhof et al. 2001, p.211). The remaining problem that can not be solved is to determine on which of the frequencies that the cycle slip occurred.

2.4.2.3.3 Geometry free solution

Another approach that we use in UGPS to detect cycle slips is the geometry-free solution, (Leick 2004, p.244). In this approach, all observations are combined into a solution, which in matrix notation becomes:
where $\Delta$ is a parameter which includes the clock errors, and the tropospheric error and $\alpha_f = (\lambda_1/\lambda_2)^2$ is the scale factor between the L1 and L2 frequency. This equation system can be written in matrix notations as follows:

$$
\begin{bmatrix}
    p_1 - cT_{GD} \\
    p_2 - \alpha_c T_{GD} \\
    \Phi_{L1} \\
    \Phi_{L2}
\end{bmatrix} =
\begin{bmatrix}
    1 & 1 & 0 & 0 \\
    1 & \alpha_f & 0 & 0 \\
    1 & -1 & \lambda_{L1} & 0 \\
    1 & -\alpha_f & 0 & \lambda_{L2}
\end{bmatrix}
\begin{bmatrix}
    \rho + \Delta \\
    1 \\
    N_{L1} \\
    N_{L2}
\end{bmatrix} +
\begin{bmatrix}
    \delta_{P1} \\
    \delta_{P2} \\
    \delta_{L1} \\
    \delta_{L2}
\end{bmatrix} +
\begin{bmatrix}
    \varepsilon_{P1} \\
    \varepsilon_{P2} \\
    \varepsilon_{L1} \\
    \varepsilon_{L2}
\end{bmatrix}
$$

(2.161)

where $A$ is the design matrix, completely independent of the receiver-satellite geometry, therefore this solution is called the geometry-free solution. $\delta$ and $\varepsilon$ are as before the multipath and the measuring noise vector and if they are ignored a solution to the equation system becomes:

$$
X = A^{-1}L
$$

(2.163)

where all the estimated parameters in $X$, except for the ambiguities, are time dependent. The ambiguities remain constants until a cycle slip occur. Leick (2004, p.245), show that there is a high correlation between the parameters in $X$, and to solve this problem it is possible to use an additional matrix $Z$ that de-correlates the parameters as follows:

$$
z = ZX
$$

(2.164)

where $Z$ is:

$$
Z =
\begin{bmatrix}
    1 & 0 & 0 & 0 \\
    0 & 1 & 0 & 0 \\
    0 & 0 & 1 & -1 \\
    0 & 0 & 1 & 0
\end{bmatrix}
$$

(2.165)

The new parameters, which are estimated after the transformation, are:

$$
z = [\rho + \Delta \ 1 \ N_w \ N_{L1}]^T
$$

(2.166)

The new parameter $N_w = N_{L1} - N_{L2}$ is the wide-lane ambiguity, which has a standard deviation of 0.25 cycles of $I_w$ according to Sjöberg (2005, p.85) and is therefore a very good indicator of a cycle slip, but then, again, it is difficult to determine on which frequency the cycle slip occur since the wide lane ambiguity is a combination of the ambiguities on L1 and L2.

2.4.2.3.4 Predicted residuals

The last algorithm that is presented here to detect cycle slips is based on the predicted observation residuals. The predicted residuals will have a stochastic nature, if the models that
describe the dynamic change of each estimated parameter are true. If not, this will directly be
evident in the predicted residuals. A cycle slip is a typical example of such situation, and
therefore the predicted residuals are suitable to detect gross errors like cycle slips. In Figure 14
the predicted residuals are presented at a rover receiver. A small cycle slip is simulated in epoch
251 by adding a $1\lambda$ shift to the phase observations towards satellite 28.

![Predicted residuals of all observations at receiver B](image)

**Figure 14. Predicted observation residuals at receiver B**

The predicted residuals before the shift, follow each other up and down as a group. This
indicates that the used dynamic models of some of the unknown parameters are incorrect. A
similar pattern is found in the predicted residuals at each receiver, but it is unique at each of
them. Comparing this pattern of predicted residuals on the L1 and L2 frequencies at each
receiver show that they follow the same pattern, which indicates that it is caused by a parameter
that is unique at each station and influences both frequencies in the same way.

Before the predicted residuals are calculated, all deterministic parameters are removed from the
measured observations. Among these is the deterministic parameter of the receiver clocks
found, calculated with single point positioning algorithm; see Section 2.3.2. These deterministic
values are not as accurate as the corresponding estimate of the remaining clock error in the
Kalman filter, mainly since observations from only one epoch are used. These calculated
parameters will directly influence all the predicted residuals in the same way, just as the pattern
shown in Figure 14.

There are two methods that can be used to remove the fluctuations in the predicted residuals
caused by the deterministic receiver clock. The first is the mean value of the predicted residuals
of the phase observation calculated each epoch. The calculated mean value is then subtracted
from all the predicted residuals and thereby is the influence of the deterministic clock correction removed.

The second method is based on the knowledge that the receiver clock is common for all observations performed at one receiver. By subtracting two observations from each other it is possible to remove the influence of the common receiver clock. A typical example when using this approach is shown in Figure 15, where the predicted residuals of observations towards satellite 18 and 28 are subtracted from each other. The simulated cycle slip can easily be detected by some shift detection algorithm.

![Graph showing difference between predicted residuals](image)

**Figure 15. By subtracting the predicted residuals for observations to two satellites it is possible to find the cycle slip**

These methods to detect cycle slips based on predicted residuals show a great potential to detect small cycle slips. The cycle slip identification is not yet implemented in the UGPS software but one idea here is to use a CUSUM chart, (cf. Section 2.5.6).

### 2.4.3 Fill in matrices L, H and R

Some vectors and matrices (L, H and R) are filled-in each recursive loop of the Kalman filter. The size of this vector and the matrices are directly related to the numbers of receivers, satellites and frequencies that are observed. To describe the sizes of the matrices let us use \( n_{\text{REF}} \) to denote the number of reference receivers \( n_{\text{ROV}} \) the rover receivers and the total number of receivers, \( n_{\text{REC}} \) is calculated as:

\[
\begin{align*}
n_{\text{REC}} & = n_{\text{REF}} - n_{\text{ROV}} 
\end{align*}
\]
Further more we denote the number of observations of each observation type (P1, P2, L1 and L2) at a station A by $n_{A,P1}$, $n_{A,P2}$, $n_{A,L1}$ and $n_{A,L2}$. The total number of observations at station A $n_{A,\text{obs}}$ is then calculated as:

$$n_{A,\text{obs}} = n_{A,P1} + n_{A,P2} + n_{A,L1} + n_{A,L2}$$ (2.168)

and the total number of observations at all stations becomes:

$$n_{\text{obs}} = n_{A,\text{obs}} + n_{B,\text{obs}} + \cdots + n_{Z,\text{obs}}$$ (2.169)

The ordering of observations is per station, e.g. first shown is the reference station with all observed satellites, and then follows all roving stations. The observation vector for Z receivers becomes:

$$\mathbf{I}_A = \begin{bmatrix} \mathbf{I}_{A,P1}^T & \mathbf{I}_{A,P2}^T & \ldots & \mathbf{I}_{A,Z}^T \end{bmatrix}^T$$ (2.170)

where A, B,...,Z are the receivers. The observation vector for station A to z satellites are given by:

$$\mathbf{I}_A = \begin{bmatrix} \mathbf{I}_{A,1}^T & \mathbf{I}_{A,2}^T & \ldots & \mathbf{I}_{A,z}^T \end{bmatrix}^T$$ (2.171)

where each element is a matrix containing the actual observations modified with the deterministic parameters:

$$\mathbf{I}_A = \begin{bmatrix} \rho_1^{S,A} - (\rho_1^{S} + c)\delta t_{A,\text{SPP}} + c s_{c, A} + c(\delta t^S - T_{GD}^S) - T_{h, A} - l_{d, A}^S - \delta A_{A,P1}^S \\ \rho_2^{S,A} - (\rho_2^{S} + c)\delta t_{A,\text{SPP}} + c s_{c, A} + c(\delta t^S - \alpha_T T_{GD}^S) - T_{h, A} - \alpha_f l_{d, A}^S - \delta A_{A,P2}^S \\ \Phi_{A,1}^S - (\rho_1^{S} + c)\delta t_{A,\text{SPP}} + c s_{c, A} + c\delta t^S - T_{h, A} + l_{d, A}^S - \delta A_{A,L1}^S \\ \Phi_{A,2}^S - (\rho_2^{S} + c)\delta t_{A,\text{SPP}} + c s_{c, A} + c\delta t^S - T_{h, A} + \alpha_f l_{d, A}^S - \delta A_{A,L2}^S \end{bmatrix}$$ (2.172)

We have ignored the multipath $\delta M_{A,P1}^S$ and the antenna offset at the satellites in this expression. This is correct during the first 24 hours when no corrections for the multipath are determined and if ephemerides that are defined for the antenna phase centres of the satellite antennas are used (like the broadcasted ephemerides).

Predicted observations are calculated by the direct functional relationship between the predicted parameters in the state vector and the observations:

$$\mathbf{h}(\mathbf{x}) = \begin{bmatrix} \rho_1^{S,-} + (\rho_1^{S} + c)\delta t_{A}^S + T_{w, A}^S + l_{d, A}^S - \delta O_{A}^S \\ \rho_2^{S,-} + (\rho_2^{S} + c)\delta t_{A}^S + T_{w, A}^S + \alpha_f l_{d, A}^S - \delta O_{A}^S \\ \rho_1^{S,-} + (\rho_1^{S} + c)\delta t_{A}^S + T_{w, A}^S - l_{d, A}^S + \delta O_{A}^S + \lambda N_{A,L1}^S \\ \rho_2^{S,-} + (\rho_2^{S} + c)\delta t_{A}^S + T_{w, A}^S - \alpha_f l_{d, A}^S + \delta O_{A}^S + \lambda N_{A,L1}^S \end{bmatrix}$$ (2.173)

A parameter that needs further explanation is $\dot{\rho}$, which is the orthogonal projection of the satellite velocity vector onto the vector between the GPS-antenna and the satellite $\rho$. The velocity vector of the satellite $\mathbf{v}^e$ is expressed in e-frame (earth frame), obtained from standard orbits as the first derivative of the polynomial. $\dot{\rho}$ is orthogonal projection of $\mathbf{v}^e$ onto $\rho$: 
\[ \hat{\rho} = |\mathbf{v}^e| \cos \alpha \]  

(2.174)

where \( \alpha \) is the angle between \( \mathbf{v}^e \) and \( \rho \), which is the scalar product of the vectors calculated as:

\[ \cos \alpha = \frac{\rho \mathbf{v}^e}{|\rho||\mathbf{v}^e|} \]  

(2.175)

Inserting Eq. (2.175) to Eq. (2.174) we get the orthogonal projection:

\[ \hat{\rho} = \frac{\rho \mathbf{v}^e}{|\rho|} \]  

(2.176)

To describe how the design matrix \( \mathbf{H} \) is filled-in the observations at two stations from one satellite are used:

\[ \mathbf{H}_{n_{\text{obs}} \times n_{\text{par}}} = \left[ \begin{array}{cccc} \mathbf{H}_{\text{PV}} & \mathbf{H}_{\text{St}} & \mathbf{H}_{\text{T}} & \mathbf{H}_{\text{l}} & \mathbf{H}_{\text{O}} & \mathbf{H}_{\text{AMB}} \end{array} \right] \]  

(2.177)

where \( n_{\text{obs}} \) is the number of phase observations and \( n_{\text{SAT}} \) is the total number of satellites. Each of the components in the design matrix \( \mathbf{H} \) will be given in the following equations for a reference station A, a rover station B and one satellite. The first submatrix of Eq.(2.177) is the design matrix for the position and velocity:

\[ \mathbf{H}_{\text{PV}} = \left[ \begin{array}{c} \mathbf{H}_{\text{A,PV}} \\ \mathbf{H}_{\text{B,PV}} \end{array} \right] = \left[ \begin{array}{cccc} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ a_x & a_y & a_z & 0 & 0 \\ a_x & a_y & a_z & 0 & 0 \\ a_x & a_y & a_z & 0 & 0 \\ a_x & a_y & a_z & 0 & 0 \end{array} \right] \]  

(2.178)

When positioning with the P-model all columns in the design matrix \( \mathbf{H} \) related to the velocity will be removed. Thereafter follows the design matrix for the receiver clocks:

\[ \mathbf{H}_{\text{St}} = [\mathbf{H}_{\text{A,St}} \quad \mathbf{H}_{\text{B,St}}]^T \]

\[ = \left[ \begin{array}{cccc} (\rho_A^5 + c) & (\rho_A^5 + c) & (\rho_A^5 + c) & 0 & 0 & 0 & 0 \end{array} \right] \]  

(2.179)
The design matrix for the tropospheric delay is given by:

\[
H_T = \begin{bmatrix}
    m_n(e^i) & m_n(e^j) & m_n(e^k) & m_n(e^l) & \frac{d^{-1}}{\sum d^{-1}} & m_n(e^m) & \frac{d^{-1}}{\sum d^{-1}} & m_n(e^n) & \frac{d^{-1}}{\sum d^{-1}}
\end{bmatrix}^T
\]

(2.180)

where \(d\) are the distances between the stations. In the case of two receivers, \(d^{-1}/\sum d^{-1} = 1\), but not if there are more reference stations. An example of the design matrix of common errors is:

\[
H_{St} = \begin{bmatrix} H^S_{A,o} & H^S_{B,o} \end{bmatrix}^T = [1 \ 1 \ 1 \ 1 \ 1 \ 1 \ 1]^T
\]

(2.181)

in the case of one satellite and two receivers.

The design matrix for the ionosphere is given by:

\[
H_i = \begin{bmatrix}
    1 & \alpha_f & -1 & -\alpha_f & \frac{d^{-1}}{\sum d^{-1}} & \alpha_f & \frac{d^{-1}}{\sum d^{-1}} & -\alpha_f & \frac{d^{-1}}{\sum d^{-1}}
\end{bmatrix}^T
\]

(2.182)

and finally the design matrix for the ambiguities becomes:

\[
H_{AMB} = \begin{bmatrix}
    0 & 0 & 0 & 0 \\
    0 & 0 & 0 & 0 \\
    \lambda_1 & 0 & 0 & 0 \\
    0 & \lambda_2 & 0 & 0 \\
    0 & 0 & 0 & 0 \\
    0 & 0 & 0 & \lambda_1 \\
    0 & 0 & 0 & \lambda_2 
\end{bmatrix}
\]

(2.183)

Figure 16 shows the general overview of the design matrix when 8 satellites are present at the reference station and 7 at the rover station. Each dot in the plot represents an non-zero value.

The weight matrix \(R\) is a square matrix of size \(n_{obs} \times n_{obs}\), where the diagonal element contains the weight of each individual observation. The actual weight for each observation depends on the chosen weighting method; see Section 2.4.2.2.1.

### 2.4.4 Quality control

The quality control in the UGPS software is performed in the Kalman filter loop before the update step by use of the predicted residuals \(\mathbf{v}_k\) and the corresponding covariance matrix \(Q_{v_k}\), using the algorithms for the local overall and local individual tests given in Section 2.1.3.3. The quality control comes late, in a computational point of view, since all matrices already are generated at this point and if a cycle slip is detected must all the matrices and vectors be redefined without the erroneous observations. It would be better if the control algorithm were
implemented before the pre-processing step when no matrices are generated, but since the design matrix $H_k$, generated after the pre-processing step is needed in the covariance prediction, see Eq. (2.43), this cannot be implemented earlier. It is possible to overcome this problem by implementing separate control algorithms before the real Kalman filter loop, but the UGPS software does not follow this approach, so it is left for further research.

![Design matrix layout for a dual-frequency situation. At the reference receiver A observations are made towards 8 satellites and at the rover 7 satellites.](image)

The local overall test is used to check if the dynamic model is correct, but it will also signal if there are any errors within the observations. Typical failures in the dynamical model are cycle slips and abrupt displacements of the roving receivers. The reason that the local overall test fails in these cases are simply that the model is not designed to compensate for these abrupt changes. To find out if a failure in the overall test is caused by a simple cycle slip or displacement follows individual local tests of each observation. A displacement could be highlighted if several observations fails the individual test at the same time, but one should have in mind that this result could also be caused by several cycle slips.
If one observation fails the individual test, we reset the ambiguity parameter just as in the case of cycle slip detection. Approach allows the filter to determine new value of the ambiguities. An alternative approach is to estimate the size of the cycle slip and correct the estimated values in the state vector with the values in a correction step used by Teunissen (1990). Our approach might release the ambiguities more frequently, but this is reasonable since the undifferenced approach is sensible to undetected cycle slips. If all observations at a roving receiver fail the individual cycle slip test, it is most likely that a displacement of the receiver has occurred and no further action is taken to correct the observations, since the fault in this case depends on the model and not the observations.

In Section 2.4.2.3.4 we noticed that there is a systematic influence in the predicted residuals caused by the estimated deterministic part of the receiver clock. This systematic error has to be removed before the hypothesis testing is performed and is made at each receiver by subtracting the mean value of the predicted residuals (of the phase observations at one receiver) from all predicted residuals.

2.4.5 Ambiguity fixing

To reach high accuracy position using phase observations it is necessary to determine the unknown ambiguities. Many different ambiguity fixing methods have been developed during the years; among them the Fast Ambiguity Resolution Approach (FARA) by Frei and Beutler (1990), the Least-Squares AMBiguity Decorrelation Adjustment (LAMBDA) by Teunissen (1994) and the "KTH-method" presented by Sjöberg (1997), (1998a), (1998b), (1999), Sjöberg and Horemuž (1999) and Horemuž and Sjöberg (2002). The goal with all of them is to find the optimal combination of a fixed set of phase ambiguities $x$ that minimizes the residual between the fix and the float phase ambiguities $\bar{x}$:

$$
\min = (\bar{x} - x)^T Q_x^{-1} (\bar{x} - x)
$$

where $Q_x$ is the covariance matrix of the float solution.

One of the most successful methods is the Lambda method developed at Delft University of Technology and introduced by Teunissen (1994) and described by Jonge and Tiberius (1996). The LAMBDA-method has become a standard within GPS surveying since it gives a good result and is time efficient. The time efficiency depends mainly on the low number of candidates of the fix solution that the method produces. The input variables to the method are the float solutions of the ambiguities that are estimated in a least squares estimation and the corresponding covariance matrix. The Lambda-method consists of three essential steps: in the first step the covariance matrix of the float solution is decorrelated to reduce the search space. In the second
step all possible candidates are determined, and in the final step the ambiguities are fixed to integer value.

We use a Matlab algorithm for the Lambda method that is developed by Joosten (2001) converted into C++. The input parameters that we use are the ambiguity parameters in the state vector $X_k$ and the corresponding part of the covariance matrix $Q_x$. The algorithm returns two alternative solutions of the ambiguities, the best and the second best solution. To confirm the strength of the solution, we use the same approach as many other software packages, (Leick 2004 p.293) and controls the ratio between the best and second best solution from Eq.(2.185) as:

$$\frac{\text{min}_{2nd\ best}}{\text{min}_{best}} > 3$$  

Eq.(2.185)

where $\text{min}_{2nd\ best}$ and $\text{min}_{best}$ are the quadratic forms of the second best and the best solutions. If the ratio between the solutions fulfils the condition in Eq (2.185), the ambiguities are fixed, otherwise they are kept unfixed until the next epoch, where a new ambiguity fixing algorithm is performed.

### 2.4.6 Output parameters

The UGPS software is developed to evaluate the undifferenced positioning approach for displacement detection, and therefore is it possible to store and output other parameters than the coordinates. The following parameters are stored each recursive epoch in UGPS:

- Estimated parameters in the state vector $X$ together with their standard deviation (determined as the square root of the diagonal elements in the covariance matrix $Q_x$)
- The cycle slip detection parameters for single-frequency combination, iono-free combination and the geometry-free combination
- The quality indicators that can be used for observation weighting: elevation angle and $C$/No value
- Number of satellites at each receiver
- Coordinates that are transformed into a local coordinate system, with origin in the initial position of the receiver
- The residuals for each observation type calculated as $v = \hat{L} - h(\hat{X})$
- The predicted residuals for each observation type calculated as $v = \hat{L} - h(\hat{X}^-)$
- The overall quality control parameter $T_k$

### 2.5 Automatic Displacement Monitoring Algorithms

An essential part of a real-time displacement monitoring system, beside from the actual positioning algorithms, is the alarm algorithms used to alarm that a point has moved out of its
position. In this section we theoretically introduce different algorithms to detect abrupt shifts in a process mean. The practical implementation and testing of the methods is performed in Chapter 3.

A typical displacement that we try to detect is when a GPS-antenna suddenly moves from one position to another. To reduce the extent of the investigations in this thesis we do not consider other motion patterns like periodical and slow motions linear motions. Research within this subject can be found in Ogaja (2002). Figure 17 shows a typical example of the type of shift that we try to detect. A vertical displacement occur at epoch \( k \), before the shift is the vertical position normally distributed as \( N(\mu_0, \sigma^2) \) and after the shift \( N(\mu_0 + \Delta, \sigma^2) \) where the shift size is represented by \( \Delta \).

![Vertical displacement](image)

**Figure 17. A vertical displacement occurs in epoch \( k \), but it is unknown in time the epoch \( k \) when the vertical displacement occur is unknown**

This can also be written in equation form as:

\[
\begin{align*}
\mu_1 &= \mu_0 + \Delta & \text{if } k < t \\
\mu_1 &= \mu_0 + \mu_1 & \text{if } k \geq t
\end{align*}
\]

where \( \mu_1 \) is the mean value.

This section focus on different approaches that are used in manufacturing processes for quality control that already has been shown useful for displacement monitoring by other researches, like Mertikas (2001), Ogaja (2002) and Mertikas and Damianidis (2007). The aim here is to give an overview of the methods and to study their performance within the undifferenced positioning approach, that is presented in this thesis.
2.5.1 Design issues

Alarm, isolation and adaption are the three main problems concerning the design of a failure detection algorithm according to Willsky (1976) and Tunissen (1990). The alarm is a pure binary decision. The decision is taken with use of a hypotheses test where the zero-hypothesis is that no displacement has occurred and the alternative that it has occurred. The idea is that the system should give an alarm as soon the process has gone out-of-control. After an alarm it is necessary to isolate the fault. This is normally made with a set of alternative hypothesis related to the possible causes of the shifts e.g. when the coordinates in one of the stations has changed or if a cycle slip is detected. And finally an adaption algorithm is needed, when the alarm has gone off and the fault is isolated, to remove the influence of the error. In the case of a cycle-slip the shift size should estimated and the state vector updated with the correct ambiguity value.

Willsky (1976) highlights that a quality control system should respond as quickly as possible and send out an alarm after a shift has occurred. A measure of the performance of a system to alarm quickly is the average-run-length (ARL), which is a measure of the average number of observation epochs (or time) from the actual shift until the system detects it and sends out an alarm. Two different types of ARL values are normally determined for each control charts; the in-control ARL and the out-of-control ARL. The in-control ARL describes the number of epochs between two incorrectly signalled alarms, and the out-of-control ARL the number of epochs it takes before an out-of-control shift is signalled. A well performing failure detection system has a long ARL in-control ARL and a short out-of-control ARL. A direct parallel can be found by the Type I and II errors in traditional hypothesis testing. Type I errors represents the risk of incorrectly rejecting a correct value and Type II the risk of incorrectly accepting a value that actually contains an error, see (Koch, 1999). It will be shown later in this chapter that a system that responds quickly also is more sensible and will send out more false alarms. This contradiction has to be treated with care when designing a failure detection system. A detector system is said to be optimal if, for a fixed mean time between two false alarms the detection time is minimised, (Basseville and Nikiforov, 1993).

Statistical Process Control (SPC) has been an issue of manufacturing since the beginning of serial production. The main purpose with SPC is to detect when a process is running as it should “in-control” or if it is “out-of-control”. During in-control the process is running within normal variations and during out-of-control the process is operating in presence of some assignable cause. Several different SPC-tools are developed over the years to follow the quality of a manufacturing process. Within SPC the magnificent seven forms the framework for quality control and improvement, which consists of the following tools, Montgomery (2001, p.153):
1. Histogram
2. Check sheet
3. Pareto chart
4. Cause-and-effect diagram
5. Defect concentration diagram
6. Scatter diagram
7. Control charts

Histogram, Scatter diagram and Control charts are the most important tools among the magnificent seven when designing algorithms for automatic displacement detection. Histogram and scatter diagrams are used for pattern recognition of the parameter distribution and the control charts used to detect if the process is running in- or out-of-control. The other tools, among the magnificent sevens, are more related to quality control in manufacturing processes than in displacement detection, so they will not be studied further.

The basic design of a control chart consists of a chart with three horizontal lines. A Centre Line (CL) that represents the normal value of the process and two lines, one above and one below, that symbolizes the upper control limit (UCL) and the lower control limit (LCL). The algorithm is then quite simple, time-ordered observation values are then plotted in the chart. A process with observation values that are running between UCL and LCL is considered as an in-control process. As soon one of the limit lines is crossed enters the process an out-of-control state. The process can be looked upon as a continuous hypothesis testing with the zero hypotheses that everything is running in control and the alternative hypothesis that the process is out of control. The zero hypotheses are rejected if any value gets outside the control limits.

A typical example of a control chart is found in Figure 18 where the process is running in control until epoch 7 where the process value passes the upper control limit and the process fall out-of-control.

**Figure 18. Control chart running in-control up until epoch 7 where it goes out of control**
This type of control charts were introduced by Walter A. Shewhart in the beginning of the 1930s and are thus also known as Shewhart charts, Hawkins and Olwell (1998). These are useful when introducing the concept of SPC because of its simplicity and it is has a good performance to detect large shifts in a process. To detect smaller shifts, one should use cumulative sum charts (CUSUM) or exponentially weighted moving average (EWMA) charts. A good overview of the CUSUM charts are given by Hawkins and Olwell (1998), and the corresponding of the EWMA is found in Montgomery (2001).

Control chart is normally designed according to the following 4 step scheme:

1. First specify the performance to be expected from the chart by setting an in-control ARL-value
2. Define what is to be the smallest magnitude of the shift $\delta$ to be detected as quick as possible
3. Determine LCL and UCL limits
4. Finally determine the out-of-control ARL for the given shift size $\delta$.

Common in all mentioned SPC methods is the assumption that all observations are independent and normally distributed. In reality this is not always the case, so to be able to use them we need some method to decorrelate autocorrelated observations. This is the subject of the following section.

2.5.2 Autocorrelation

All mentioned control charts take for granted a standard assumption that the observations are independent and normally distributed $N(\mu, \sigma^2)$. Observations of a process in-control can be described as follows:

$$X(t) = \mu + \varepsilon_1$$  \hspace{1cm} (2.187)

$t$ is the time, $\mu$ the mean value of the process and $\varepsilon_1$ the normally distributed measurement error with zero mean and standard deviation $\sigma$.

In reality this is not always the case, as there is more or less always some correlation between epochs in the parameters, e.g. the estimated coordinates from UGPS is correlated in time.

Imperfections in the used model of the physical reality are one reason to the correlation in time, e.g. the lack of a model for multipath. The multipath itself is autocorrelated and this together with the lack of mathematical model will result in coordinates that are correlated in time. The troposphere and ionosphere are also factors that as well as the multipath is correlated in time that will influence the result in the estimations if their influence is not completely removed before adjustment.
Correlation in time series is in signal processing called autocorrelation, defined as:

\[ R_X(t_i, t_{i+k}) = E\{X(t_i)X(t_{i+k})\} \]  \hspace{1cm} (2.188)

Here \( t_i \) is the time of the actual epoch and \( k \) constitutes the lag between the epochs, and \( E\{\} \) is the statistical expectation operator. The correlation between two epochs is given by the auto-covariance function \( C_X(t_i, t_{i+k}) \) given by:

\[ C_X(t_i, t_{i+k}) = E[X(t_i) - \mu][X(t_{i+k}) - \mu] \]  \hspace{1cm} (2.189)

These two functions can be found within any textbook about adaptive filter theory e.g. Brown and Hwang (1997) of Sayed (2003). By normalising Eq (2.189) we get the correlation function used to calculate a value of the autocorrelation \( \rho_k \) for a lag \( k \)

\[ \rho_k = \frac{C_X(t_i, t_{i+k})}{R_X(t_i, t_i)} = \frac{C_X(t_i, t_{i+k})}{\sigma_i^2} \]  \hspace{1cm} (2.190)

The autocorrelation factor has values between 0 and 1. If no lag occurs \( k = 0 \) then the sample autocorrelation is equal to 1 since the covariance function in the nominator is exactly the same as the variance in the denominator. The autocorrelation will decrees to 0 when the lag \( k \) becomes larger. To compute the autocorrelation factor for a sample of observations we use:

\[ r_k = \frac{\sum_{i=1}^{n-k}(x(t_i) - \bar{x})(x(t_{i-k}) - \bar{x})}{\sum_{i=1}^{n-k}(x(t_i) - \bar{x})^2} \]  \hspace{1cm} (2.191)

where \( n \) is the sample size, \( \bar{x} \) the mean value of the sample, \( i \) the actual epoch number. To estimate the sample autocorrelation factor \( r_k \) about 1/4 of the total sample size \( n \) is needed Montgomery (2001, p.462).

### 2.5.2.1 Methods to remove autocorrelation

The Control chart performance depends on the autocorrelation in the time series. The optimal situation is when no autocorrelation between the observations is present. There are several different useful approaches to reduce the influence of autocorrelation. One is to use the knowledge that correlation between observations reduces with time. This implies that the correlative structure in a time series can be reduced by increasing the time lag between epochs. Another approach is to model the correlative structure with an appropriate model and use this model to create a new set of observations free from autocorrelation. The final alternative is to adjust the control limits in the control charts to compensate for the autocorrelation. More details concerning this method can be found in Lu and Reynolds (2001). Here we study the two first alternative methods.
2.5.2.1.1 Increase the observation lag

In the introduction of this section it was mentioned that the autocorrelation between observations usually decreases with time. This implies that to reduce correlation between successive observations in a time series it is possible to increase the time interval between the observations until there is no appreciable correlation between the observations. Correlation and scatter plots are useful tools to find the time interval for an uncorrelated time series.

The procedure implies that the number of observations is reduced by removing some of the observations, which implies that some information is lost. This is the reason why some authors, as Montgomery (2001) and Hawkins and Olwell (1998), do not recommend this approach. On the other hand Ince and Sahin (2000) accept these performance drawbacks and use this approach in their displacement monitoring system RT-MODS2, developed at Istanbul Technical University.

2.5.2.1.2 Model the correlative structure

An alternative approach to reduce the influence of the autocorrelation is to construct a model that describes the correlative structure between the observations in a dataset and then use it to generate a new dataset with observations free from correlation.

One approach is to use the residuals or “innovations” between the real observations in epoch \( t \) and predicted observations generated with the model that takes care of the autocorrelated structure. The resulting residuals are assumed to be independent and normally distributed and can be applied in the control charts. The predicted target values are determined by adapting a model to preliminary data or by using a forecast model like the Exponentially Weighted Moving Average (EWMA); see section 2.5.4.

One frequently used model for this purpose is the Box-Jenkins autoregressive-moving average (ARMA) model, here given without the Moving Average part as a pure autoregressive AR-model, Hawkins and Olwell (1998, p.220). That expresses the time series \( X_t \), as:

\[
X_t = \mu + \phi_1 X_{t-1} + \phi_2 X_{t-2} + \cdots + \phi_p X_{t-p} + e_t
\]  

(2.192)

where the subscript \( t \) indicates an epoch in time. There are \( p \) autoregressive parameters \( \phi_i \) connecting \( X_t \) with its \( p \) most recent predecessors. The values of the autoregressive terms are between \((-1 < \phi_i < 1)\). Further, \( e_t \) is an independent and normally distributed \( N(0, \sigma_e) \) error term, \( \mu \) a constant and \( \mu \) the mean value of the \( X_t \) series. The AR-model describes a process with stationary behaviour, where the observation value \( X \) wanders around a fixed mean value. Reducing the recursion in this model to one step, we get the first-order autoregressive model \( AR(1) \):
\[ X_t = \frac{(1 - \phi)}{\Delta} \mu + \phi X_{t-1} + \epsilon_t \quad (2.193) \]

where the subscript of the autoregressive parameter \( \phi \) is dropped since there is the only one term left in the expression. The AR(1) process could be written as:

\[ X_t - \mu = \phi (X_{t-1} - \mu) + \epsilon_t \quad (2.194) \]

With a preliminary dataset free from any abrupt shifts it becomes possible to estimate the unknown parameter \( \phi \) in Eq. (2.193) with a LSQ-adjustment using Eq.(2.194) as the observation equation. Once estimated the parameter \( \hat{\phi} \) it is easy to convert a correlated time series \( X_t \) into an uncorrelated normal distributed time series \( Z_t \) as:

\[ Z_t = X_t - \hat{\phi} X_{t-1} \quad (2.195) \]

The estimation of the constant parameters has to be done with care, since small undetected errors in the preliminary dataset will influence the performance of the control chart.

Hawkins and Olwell (1998) analyses an in control situation for this process and the reasons to out-of-control situations. The different cases can be summarised as:

- The mean of \( X_t \) could go from a constant level \( \mu_t \) to another constant level \( \mu_t + \Delta \) as a step change
- The mean of the residuals \( Z_t \) could go from zero to some non-zero level \( \delta \)
- The variance of the innovation series could go from \( \sigma_0^2 \) to some other value \( \sigma_1^2 \)

The step-changes in mean \( X_t \) and \( Z_t \) do not lead to the same outcome. If a step-change \( \Delta \) occurs in the mean of a set of observations \( X_t \), there will be a spike in the decorrelated observations \( Z_t \) with the height \( \Delta \). The mean of each subsequent decorrelated observation in \( Z_t \) will change into:

\[ \delta = \Delta (1 - \phi) \quad (2.196) \]

Thus a small correlation parameter \( \phi \) will result in a small spike and a large step-change, and if \( \phi \) is close to 1 the shift will occur as a spike large spike followed by a small step-change. A typical high correlation situation is found in Figure 19, which shows a decorrelated observation set \( Z_t \), decorrelated with \( \phi = 0.9607 \). The spike, in epoch 3000 is caused by a \( \Delta = 0.05 \) m step change in the original dataset \( X_t \). The height of the spike is the same as the step change and in the following epochs changes the mean of \( Z_t \) to \( \Delta (1 - \phi) = 0.002 \) m.

If the mean of \( Z_t \) changes from 0 to \( \delta \) at some instant then the mean of \( X_t \) at time \( m \) units later will change from \( \mu \) to:

\[ \mu_m = \mu_0 + \delta \sum_{j=0}^{m} \phi^j = \mu_0 + \delta \frac{1 - \phi^{m+1}}{1 - \phi} \quad (2.197) \]
where \( \mu_0 \) is the mean value before the shift and \( \mu_z \) after the shift. Equation (2.197) shows that a change \( Z_t \) does not correspond to step-change in the observation \( X_t \).

\[
\Delta \approx \Delta (1 - \phi)
\]

Figure 19. Decorrelated observations with a 5cm shift in epoch 3000

The shape of the shift and the time until \( X_t \) has reached a new stable value is directly influenced by the correlation parameter \( \phi \). A small correlation will result in a quick adaption to the new value and large correlation will result in the opposite, see Figure 20. The mean value \( X_t \) of after the shift in \( Z_t \) can be derived to \( \mu_0 + \delta / (1 - \phi) \) by letting \( m \) in Eq.(2.197) go to infinity.

Figure 20. Shift in the mean value \( \varepsilon(t) = 0.010 \) m is introduced in epoch 1. The figure show how different correlation coefficients (0.1, 0.5 and 0.9) influences the estimated values of the process \( X(t) \)
It is necessary to have these properties of the decorrelated observations in mind when an automatic shift detection algorithm is created.

2.5.3 Shewhart charts

Shewhart charts are groups of charts based on sets of random observations that are gathered in time ordered groups or samples \( X_k \) (\( k = 1,2,3, \ldots \)), each with \( n \) observations. All samples are assumed to be independent and uncorrelated, if they are not, it is possible to use of the methods in the previous section.

Several different types of Shewhart charts are used to detect shifts in a time series. The Xbar-chart is used to detect shifts in the mean value in a time series, the R-chart to monitor the range between the maximum and minimum value of the observations in a sample and the Sbar-chart to follow the sample variance. R-chart can be seen as a precursor of the Sbar-charts and work reasonably well under some special circumstance i.e. when the sample size \( n \) is small. For this reason the R-chart is left here and will not be considered further in these theses.

The properties of each sample are statistically tested in the Shewhart charts with a traditional hypothesis tests against limits that are determined form known values of the process, as the mean value and the standard deviation. Since these values seldom are known one has to estimate them from some preliminary dataset with independent and normally distributed observations that are free from any abrupt shifts. For example, the control limits in the Xbar-chart is the following, if the mean value and the standard deviation of the process is known from the beginning:

\[
\begin{align*}
UCL &= \mu + h \frac{\sigma}{\sqrt{n}} \\
CL &= \mu \\
LCL &= \mu - h \frac{\sigma}{\sqrt{n}}
\end{align*}
\]  

(2.198)

Here \( h \) is a value normally 2 or 3 that determines the confidence level that represents the control limits. The control limits of a Shewhart chart are directly dependent of the estimated mean value and standard deviation, this implies so the estimation of these parameters has to be done with care. If e.g. the standard deviation is incorrectly estimated with a too small value, then when the process is running there will be a lot more false alarms since the confidence interval that are spanned by UCL and LCL are too narrow. The opposite will occur if the standard deviation is estimated as a too high value become, then the chart would not alarm correctly when the process runs out-of-control. In the case the mean value is incorrectly estimated, the process will run too close to either the UCL or LCL and cause too many out-of-control situations on that side of the mean value that is closest to one of the limit values and too few out-of control values on the opposite side in the control chart since the distance to the control limit is too large.
To estimate the mean value and the standard deviation, Montgomery (2001) proposes the following approaches. The estimation of the mean value $\bar{X}$ is calculated straightforward as the mean of the mean values $\bar{X}$ of the $m$ subgroups of size $n$ of the dataset:

$$\bar{X} = \frac{\bar{X}_1 + \bar{X}_2 + \bar{X}_3 + \ldots + \bar{X}_m}{m}$$  \hspace{1cm} (2.199)

where the mean of each sample is calculated as:

$$\bar{X} = \frac{1}{n} \sum_{i=1}^{n} X_i$$  \hspace{1cm} (2.200)

A similar procedure using subgroups is followed when estimating the standard deviation from the preliminary dataset:

$$\bar{S} = \frac{S_1 + S_2 + S_3 + \ldots + S_m}{m}$$  \hspace{1cm} (2.201)

where the standard deviation of each sample is calculated as:

$$S = \sqrt{\frac{\sum_{i=1}^{n} (X_i - \bar{X})^2}{n-1}}$$  \hspace{1cm} (2.202)

It is necessary to be aware of that this estimation of the standard deviation is not an unbiased estimation of $\sigma$. If the size of the subgroups are small will the underlying distribution is normal will $S$ not estimate $\sigma$. A constant $c_4$ can then be used to scale $S$ into $\bar{S}$. The $c_4$ constant depends on the actual sample size $n$. The constant can approximately be calculated as:

$$c_4 = \frac{4(n-1)}{(4n-3)} \text{ for } n \geq 25$$  \hspace{1cm} (2.203)

For smaller values of the group size one should use the tabular values in Montgomery (2001, Appendix VI). $c_4$ will converge towards 1 when $n \to \infty$.

This gives us new and more correct control limits for the Xbar chart:

$$UCL = \bar{X} + h \frac{\bar{S}}{c_4 \sqrt{n}}$$

$$CL = \bar{X}$$

$$LCL = \bar{X} - h \frac{\bar{S}}{c_4 \sqrt{n}}$$  \hspace{1cm} (2.204)

where $h$ is the similar confidence level as used in traditional hypotheses testing. The similar control limits for the Sbar-chart is give by Montgomery (2001 p.241) as:

$$UCL = \bar{S} + h \frac{S}{c_4 \sqrt{1 - c_4^2}}$$

$$CL = \bar{S}$$

$$LCL = \bar{S} - h \frac{S}{c_4 \sqrt{1 - c_4^2}}$$  \hspace{1cm} (2.205)
It is possible to optimise the Shewhart charts to detect a certain shift size $\delta$. Hawkins and Olwell (1998, p.44) give the following relation:

$$n = \left(\frac{6}{\delta}\right)^2$$  \hspace{1cm} (2.206)

where $n$ is the number of observations in each sample and $\delta$ is the shift given in standard deviations of the process values. Assume that the minimum shift we want to detect is $2.5\sigma$. Then the sample size $n$ becomes $5.76 \approx 6$ observations. If the used sample size in the Shewhart chart is $n = 1$, the minimum detectable shift becomes $6\sigma$.

The in-control Average Run Length (ARL) of a Shewhart chart depends on the chosen confidence level. This will be shown in the following example: Assume that at the confidence interval level is set $h = 3\sigma$. The probability $p$ to fall outside this interval is approximately 0.0027. The ARL is calculated as:

$$ARL = \frac{1}{p}$$  \hspace{1cm} (2.207)

which gives an approximate in-control ARL of 370 epochs, Roberts (1959). It is possible to adjust the value of $h$ to increase the ARL, but at the same time one increases the risk of accepting an incorrect value (Type II error).

The Shewhart charts are very simple and hold no memory about the previous samples. The values that are checked in the charts only depend on the sample size $n$, which is one of the parameters that can be used to change the performance of the charts. Well calibrated Shewhart charts are capable to find shifts of the size of $2\sigma$ or higher if a sample size of $n = 4, 5, \text{ or } 6$ are used (Montgomery, 2001, p.223). Smaller shifts are more difficult to find since, the sample size then has to be increased, and that increases the risk that the actual shift occur within the sample and smoothed out when the mean value is determined.

### 2.5.4 Weighted Moving Average charts

An alternative method used to overcome the delay introduced in Shewhart charts, when grouping observations into samples, is to calculate a moving average (MA) value of the $n$ most recent observations. Sparks (2004a) gives a general expression of this procedure as:

$$\bar{X}_{t,n,p} = \frac{\sum_{j=t-n+1}^{t} p_j X_j}{\sum_{j=t-n+1}^{t} p_j}$$  \hspace{1cm} (2.208)

as the sum of multiplications of the observations $X_i$ and their weights $p_i$ of the $n$:th last epochs divided by the sum of all weights. Using weights as in Eq(2.208) the moving average chart is called a weighted moving average chart (WMA).
The performance of a WMA chart is determined by the number of observations \( n \) that are used and by their weights. Usually, the expression filter memory is used when talking about the properties of a WMA filter. A large \( n \) value indicates that the filter has a large memory since a large number of observations are used to calculate the average value and with a small \( n \) the memory will be short. A chart with a large memory will react slowly on large shifts in the mean value but it will be able to detect small shifts. Thus, with a small value of \( n \) large shifts can be detected.

The weights will also influence the memory of a WMA chart. The simplest type of weighting scheme is giving all observations equal weight, but it is more reasonable to give observations near the actual epoch \( t \) a higher weight, because the correlation between observations decline with time. Two useful approaches is the linearly and the exponential decaying weight schemes.

The weights in a linearly weighted observations can be calculated as \( p_j = j - t + n \) and for exponentially weighted observations as \( p_j = \lambda^{t-j} \) for \( j = t - n + 1, \ldots, t \). Sparks (2004a) gives the following upper and lower control limits for equally weighted observations:

\[
\begin{align*}
\text{UCL} &= \mu + h_n \frac{\sigma}{\sqrt{n}} \\
\text{LCL} &= \mu - h_n \frac{\sigma}{\sqrt{n}}
\end{align*}
\]

These control limits have to be adjusted when linearly weighted observations are used:

\[
\begin{align*}
\text{UCL} &= \mu + \sigma h_n \sqrt{\frac{2n + 1}{3n(n + 1)}} \\
\text{LCL} &= \mu - \sigma h_n \sqrt{\frac{2n + 1}{3n(n + 1)}}
\end{align*}
\]

where \( h_n \) is the control limits for the actual sample size.

One of the benefits of this method is that it is easy to implement, and everyone that is familiar with the concept of mean values can use it and easily understand the result. The main disadvantage of this method is called the problem of "inertia". It occurs when a process for some reason is running in-control close to one of the control limits in the chart, and a shift occurs in the opposite direction without being detected. Figure 21 displays this scenario, where the process value is close to LCL in epoch \( t \). In the successive epoch an upward shift occurs. The filter is not able to send out an alarm signal even if its size is much larger than the allowed shift (distance between CL and UCL), since the new value is within the control limits of the chart.
Figure 21. The problem of inertia, CL is the centre line, UCL upper control limit and LCL lower control limit.

Values of the ARL for WMA charts is determined by simulation see Sparks (2003).

2.5.5 Exponentially Weighted Moving average

Roberts (1959) introduced a process control chart similar to the Exponentially Weighted MA chart (EWMA). His basic idea is to give weight to the current epoch relative to past sampling points by a weighting parameter $\lambda$. The basic recursive algorithm is given by Montgomery (2001 p.426) like:

$$E_t = \lambda X_t + (1 - \lambda)E_{t-1}$$

with start value in epoch $t = 0$ as $E_0 = \mu_0$. The weighting parameter $\lambda$ has a value ($0 \leq \lambda \leq 1$). The size of this parameter will directly influence the chart properties. A small value of $\lambda$ is more effective than a Shewhart chart to detect small shifts, but choosing a small value will also give the chart a long reaction time when large shifts occur, since the large shift will be smoothed by the in-control data obtained before the shift. When $\lambda$, on the other hand, is large the EWMA will be more sensible to large shifts and in the extreme, when $\lambda = 1$, the EWMA chart will turn into a Shewhart chart, which have no relation to the previous epoch data. Montgomery (2001, p. 431) stated from their experience that a value of $\lambda$ between $0.05 \leq \lambda \leq 0.25$ works well in practice.

Upper and lower control limits for EWMA is given by Roberts (1959):

$$\text{UCL} = \mu + \sigma h \sqrt{\frac{\lambda (1 - (1 - \lambda)^{2t})}{(2 - \lambda)}}$$
$$\text{LCL} = \mu - \sigma h \sqrt{\frac{\lambda (1 - (1 - \lambda)^{2t})}{(2 - \lambda)}}$$

with $\mu$ and $\sigma$ as mean value and standard deviation of the process and $h$ the value that determines the in- and out-of-control ARL properties. The simulation approach is the general method to determine ARL values for EWMA charts; for details see Crowder (1987). The upper
and lower control limits will converge towards $\mu \pm \sigma h\sqrt{\lambda/(2 - \lambda)}$ when $t$ goes to infinity provided that $\lambda < 1$.

One of the benefits with the EWMA charts is that they are not as sensitive to non-normally distributed observations as the Shewhart chart. However, they are not free from problems e.g. they suffer from the same type of inertia problem as the WMA charts, especially when $\lambda$ is small. For large $\lambda$ values, the chart will act as a Shewhart chart, which does not suffer from the inertia problem, but without memory it is insensitive to small changes in the process data.

Several papers have been published on improving EWMA charts. For example Capizzi and Masarotto (2003) made the EWMA charts self-adaptive by introducing a variable weighting parameter $\lambda$. The new chart is called Adaptive Exponential Weighted Moving Average (AEWMA) and it adjusts the weighting parameter $\lambda$, from epoch to epoch, by using the difference between previous epoch in-control statistics $E_{t-1}$ and the actual observation $X_t$, written in epoch $t$ as

$$e_t = X_t - E_{t-1}$$

If the "error" value $e_t$ is large, the weight factor becomes large, and the chart will act more as a Shewhart chart and sensitive to large shifts. If, on the other hand, $e_t$ is a small value the weighting factor is set to a small value which makes the filter sensible to small shifts in the mean. Capizzi and Massarotto (2003) tested several different weighting strategies, one of which is:

$$\lambda = \begin{cases} 
\frac{e_t + (1 - \lambda)\gamma}{e_t}, & \text{if } e_t < -\gamma \\
\lambda, & \text{if } -\gamma \leq e_t \leq \gamma \\
\frac{e_t - (1 - \lambda)\gamma}{e_t}, & \text{if } e_t > \gamma
\end{cases}$$

The extra parameter $\gamma$ is used to tune the weighting performance. This parameter makes the AEWMA act as a EWMA when $e_t < \gamma$ and when $\gamma \ll |e_t|$ the chart will turn into a Shewhart chart, which is the cure for the inertia problem.

Another paper that discusses improvements of the EWMA is written by Reynolds and Stoumbos (2006). They study the performance of different combinations of EWMA and AEWMA charts on and concluded that most important factors when designing an optimal chart to simultaneously both large and small shifts is to combine charts of $\mu$ and $\sigma$. The actual choice of chart type will also influence the result, but the general improvement is found when combining $\mu$ and $\sigma$ charts.

The ARL value for a EWMA chart is determined through repeated simulations in a similar procedure as for the MA charts, Reynolds and Stoumbos (2006).
2.5.6 CUSUM charts

An alternative method to detect small abrupt shifts in tabular data collected in time series are charts based on cumulative sums (CUSUM). Page (1954) was the first to introduce this concept. The statistical model of CUSUM charts are based on the same properties as the Shewhart charts:

- Statistically independent observations, that
- Assume normal distribution, and
- The mean and standard deviation are known quantities

This implies that if the observations are autocorrelated, they have to be decorrelated with one of the methods mentioned in Section 2.5.2.1 before they can be used in a CUSUM chart.

We will give a general overview of the CUSUM chart in this section, detailed derivation can be found in Mertikas and Rizos (1997) and (Hawkins and Olwell, 1998). In a practical point of view, several researchers have used CUSUM charts in displacement detection algorithms like, Ogaja (2001), Mertikas (2001) and recently also Mertikas and Damianidis (2007).

2.5.6.1 Description of CUSUM charts

The basic equation used in CUSUM charts for monitoring the mean value of a process is the summation of the residuals between the actual observation and the known and constant mean value:

\[
C_n = \sum_{j=1}^{n} (X_j - \mu) \tag{2.215}
\]

where \( C_n \) is the cumulative sum distributed as \( N(0, n\sigma^2) \), \( n \) is the epoch number, \( X_j \) the \( j \) th reading and \( \mu \) the in-control mean value of the normally distributed process \( N(\mu, \sigma^2) \). Eq. (2.215) is used as a reference to describe a shift in the mean value \( \delta \) at epoch \( m \):

\[
C_n = \sum_{j=1}^{m} (X_j - \mu) + \sum_{j=m+1}^{n} ((X_j + \Delta) - \mu) \tag{2.216}
\]

this implies that the distribution is \( N(0, \sigma^2) \) for the process up to epoch \( m \) and thereafter \( N((n-m)\delta, (n-m)\sigma^2) \). As a result, in an out-of-control situation the average value traces centred along a path with slope \((n-m)\Delta\). In Figure 22 the output CUSUM parameter is running in-control along the first 500 epochs. In epoch 501 occurs a shift in the process mean and the CUSUM value starts to drift along a trace with the slope \((n-m)\Delta\).
Figure 22. Working principles of a CUSUM chart

Eq. (2.216) can be given in recursive form epoch \( n \) as:

\[
C_n = C_{n-1} + (X_n - \mu)
\]  

the start value of the cumulative sum is \( C_0 = 0 \).

A CUSUM chart is usually optimised for a specific size of a displacement \( \Delta \). This is made using a constant \( k \), which represents the size of the shift the CUSUM is optimised for. The constant reference value \( k \) is subtracted from \( C_n \) each iterative run as:

\[
C_n = C_{n-1} + ((X_n - \mu) - k)
\]

the outcome from the recursive algorithm will thereby be a negative value as long the CUSUM is running in-control. The CUSUM will change toward an upward direction if a shift larger than the specified value in \( k \) occurs. The relation between the reference value \( k \) and the actual shift \( \Delta \) is given by Hawkins and Olwell (1998):

\[
k = \frac{\Delta}{2}
\]

An alternative form of CUSUM is the decision interval CUSUM (DI CUSUM) defined for a positive shift as:

\[
C_n^+ = \max(0, C_{n-1}^+ + (X_n - \mu) - k)
\]

The \max() \ operator in the upward DI CUSUM takes the maximal value of the two parameters within the parentheses. When the process is running in control, becomes the right parameter negative, since the difference between \( (X_n - \mu) \) will be smaller than the reference value \( k \), and thus the largest parameter within the parentheses will be zero. When a shift occurs larger than \( k \), \( C_n^+ \) gets a positive value. If the shift is a permanent, e.g. the mean value makes a step change from one value to another, \( C_n^+ \) will continue to increase after the shift. The alarm situations for the \( C_n^+ \) CUSUM is when \( C_n^+ \) becomes higher than a threshold value \( h^+ \).

The CUSUM chart can also be defined for negative shifts as:
\[ C_n^- = \min(0, C_{n-1}^- + (X_n - \mu) + k) \]  

where the \( \min() \) operator gives the \( C_n^- \) the minimum value of the two inside the parameters. The alarm situation for the \( C_n^- \) CUSUM occurs when \( C_n^- \) passes below the threshold value \( h^- \). Positive and negative CUSUM charts are often used simultaneously as double sided CUSUMs to cover situations where both positive and negative shifts occur in the observations.

An optimal CUSUM have a large in-control ARL and a small out-of-control ARL. There are two variables that are used to optimise these ARLs for a certain shift size; the reference value \( k \), used to specify the shift size, and the threshold height \( h \), which determines the minimal alarm level. Both these variables have a direct influence on the in- and out-of-control ARL. To determine ARL one first has to determine the shift size \( \Delta \) and then use Eq.(2.219) to determine a value of \( k \). With a known \( k \) value opens two options; the first is to decide a value of the ARL, and then calculate the threshold height \( h \) based on \( \Delta \) and \( k \). The second option is to decide the threshold height \( h \) and then use this value together with \( k \) to calculate ARL. (Hawkins and Olwell 1998, pp.48-49) presents tabular values for this purpose, but they have also generated two software modules GETARL and ANYGETH that are used to calculate the unknown values.

Double sided CUSUMs can be optimised for different shift sizes. The common double sided ARL value is calculated as follows, (Hawkins and Olwell, ibid.):

\[
\frac{1}{ARL} = \frac{1}{ARL^+} + \frac{1}{ARL^-}
\]  

Where \( ARL^+ \) and \( ARL^- \) are the average run length for the upward and downward CUSUM. In the tests that we perform in this thesis we assume that the \( k \) and \( h \) have the same value for both up and down CUSUM, thus the total ARL for a double sided CUSUM is \( ARL/2 \).

One of the true benefits with the CUSUM chart is that it is simple to determine the size of a shift directly when the alarm goes off. The shift size is estimated by using the CUSUM value from the alarm epoch number \( j \) when the threshold \( h \) is passed and then values from the last in-control epoch where the CUSUM value is zero in epoch number \( j-m \), where \( m \) is the number of epochs between the CUSUM values. The shift size is then calculated as:

\[
\delta = \frac{C_{n,j} - C_{n,j-m}}{m} = \frac{C_{n,j-m}}{m}
\]  

where \( C_{n,j} \) disappear since its value is always zero.

CUSUM charts can also be defined for standardised observations. The standardisation converts the observations into a zero mean process with unit standard deviation. This type of chart is given as:
The difference between CUSUM $C_n$ and CUSUM $S_n$, is that the latter is scaled by the standard deviation $\sigma$ of the observations. This makes the CUSUMs of $C_n$ and $S_n$ identical except for the units of the vertical axis. The vertical axis of the $S_n$ CUSUM will be measured in multiples of the standard deviation $\sigma$ of the data, whereas the vertical axis of the $C_n$ CUSUM will be measured in the same units as $X$, Hawkins and Olwell (1998 p.13).

$S_n$ is the cumulative sum of the standardised observations, which has the following recursive algorithm:

\[ S_n = S_{n-1} + U_n - k_\sigma \]

where the subscript at the reference value $k_\sigma$ is used to indicates that the limits are given in multiples of the standard deviation $\sigma$ of the used dataset. The corresponding positive and negative DI CUSUM are given by:

\[ S_n^+ = \max(0, S_{n-1}^+ + U_n - k_\sigma) \]

and

\[ S_n^- = \min(0, S_{n-1}^- + U_n + k_\sigma) \]

where the start values are $S_0^+ = S_0^- = 0$. The alarm situation for $S_n^+$ CUSUM is given when $S_n^+ > h$ and for $S_n^-$ when $S_n^- < h$.

### 2.5.6.2 Influence of deviations in the statistical properties on the performance

In the opening of this section the necessary statistical properties of the observations used in a CUSUM were introduced. Here we study what happens if they do not follow the stated properties of statistically independence, normal distribution and known mean value and standard deviation.

Independent observations in a time series are free from correlation between the observations in time, they have no autocorrelation. But if this is not the case, if autocorrelation is present in the observations, how will this influence the performance of a CUSUM? (Hawkins and Olwell, 1998) studied this phenomenon and concluded that a positive correlation results in a decrease in the ARL and a negative correlation that the ARL increases. They presented the following table constructed with simulated data with various different values of $\rho$ with nominal ARL 1000.
Table 7. Influence of different values of the autocorrelation coefficient $\rho$ on the ARL for a process with ARL 1000. (Table from Hawkins and Olwell 1998 pp.77)

<table>
<thead>
<tr>
<th>$\rho$</th>
<th>$k = 1$</th>
<th>$k = 0.5$</th>
<th>$k = 0.25$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.50</td>
<td>8452</td>
<td>1408451</td>
<td>1960784</td>
</tr>
<tr>
<td>0.40</td>
<td>7148</td>
<td>146843</td>
<td>762782</td>
</tr>
<tr>
<td>0.30</td>
<td>5391</td>
<td>24149</td>
<td>17328</td>
</tr>
<tr>
<td>0.20</td>
<td>3250</td>
<td>6309</td>
<td>5333</td>
</tr>
<tr>
<td>0.10</td>
<td>1807</td>
<td>2201</td>
<td>2048</td>
</tr>
<tr>
<td>0.00</td>
<td>992.8</td>
<td>995.4</td>
<td>999.6</td>
</tr>
<tr>
<td>0.10</td>
<td>597.8</td>
<td>517.4</td>
<td>557.5</td>
</tr>
<tr>
<td>0.20</td>
<td>385.0</td>
<td>309.8</td>
<td>348.5</td>
</tr>
<tr>
<td>0.30</td>
<td>259.7</td>
<td>201.0</td>
<td>233.9</td>
</tr>
<tr>
<td>0.40</td>
<td>185.8</td>
<td>140.1</td>
<td>165.6</td>
</tr>
<tr>
<td>0.50</td>
<td>140.0</td>
<td>103.5</td>
<td>123.2</td>
</tr>
</tbody>
</table>

In the autocorrelation plots that we presented in Section 2.5.2 we found that the original data has very high autocorrelation, much higher than the values presented in Table 7. It is therefore necessary to apply some of the decorrelation algorithms presented in Section 2.5.2, page 86.

2.5.6.3 Optimising a CUSUM performance

Several different methods to improve the out-of-control detection speed of a CUSUM chart have been introduced over the year. In this section are two different approaches introduced; the fast initial response and the self starting CUSUMs.

2.5.6.3.1 Fast initial response

A known problem with the CUSUM charts is that they do not respond very quickly to shifts that occur early in a time series. By giving the CUSUM chart another start value than zero the performance of the chart to detect early shifts will increase and the out-of-control ARL becomes shorter. This is known as head start or Fast Initial Response (FIR) and corresponds to a situation where the chart already is out-of-control when it is started. Different values of the head start were studied by Lucas and Crosier (1982) and they recommend a head start of $s_0^+ = h/2$. By doing this one have to adjust the $h$ value slightly, this implies that the response time in the case when the process starts in control becomes slightly longer.

2.5.6.3.2 Self starting CUSUM

In the case when no long dataset is available to estimate the mean value and standard deviation, it is possible to use Self Starting CUSUM charts (SS CUSUM). In these charts the mean value and
The standard deviation of the process are updated each time a new observation occurs. The basic recursive algorithm of this approach is given by:

\[
\bar{X}_n = \bar{X}_{n-1} + \frac{(X_n - \bar{X}_{n-1})}{n}
\]

\[
W_n = W_{n-1} + \frac{(n-1)(X_n - \bar{X}_{n-1})^2}{n}
\]

\[
S_n^2 = \frac{W_n}{n-1}
\]

where \(\bar{X}_n\) is the mean value, \(W_n\) is the sum of squared deviations and \(S_n^2\) is the sample square variance of the first \(n\) readings. These values can be determined already after two epochs and the standardised observations can be determined according to Eq. (2.224) as:

\[
T_n = \frac{(X_n - \bar{X}_{n-1})}{S_n}
\]

were we use the updated values of the mean and standard deviation instead of \(\mu\) and \(\sigma\). With these estimated values, \(T_n\) follows a scaled t-distribution with \(n-2\) degrees of freedom. When the number of samples \(n \to \infty\) the t-distribution is assumed to converge towards normal distribution. (Hawkins and Olwell, 1998) call this an oversimplification, that works fine when comparing the mean values of two samples in a two-sample test with large samples, but not when repeated samples is used because, it ignores the correlation between the samples \(T_n\) that is introduced when using common estimates of \(\mu\) and \(\sigma\). They propose the following procedure to transform the t-distribution into a normal distribution \(N(0,1)\). The basic idea is to convert the continuous random variable into its tail area and then convert this area back to normal ordinate. This defines a new random variable, which is normally distributed. In equation form this can be written:

\[
U_n = \Phi^{-1} \left[ t_{n-2} \left( \frac{(n-1)}{n} T_n \right) \right]
\]

Where \(\Phi^{-1}\) is the inverse normal function, \(t_{n-2}\) the cumulative distribution function of Student’s t-distribution with \(n-2\) degrees of freedom. The normal distributed new parameter \(U_n\) can then be used in a SS CUSUM.

There are two types of problems that arise when using SS CUSUMs: One that occurs after a shift in the mean and the other is the sensitivity to outliers.

When a shift \(\Delta\) occurs in a dataset and a normal DI CUSUM chart is used, with known values of mean \(\mu\) and standard deviation \(\sigma\), the output starts to drift upwards along a straight line with slope \(\Delta - k\). This is not the case when a self starting DI CUSUM is used, since the mean value of the process is calculated recursively and a shift will slowly converge from the initial towards the
new level. According to Eq (2.229) and (2.230) it is obvious that a change in mean also will influence the variance of the variable. The variance is actually more sensitive to large shifts than the mean value, since the residuals between the actual observation \(X_n\) and predicted mean value \(\bar{X}_{n-1}\) is squared. The combined effect of this is that the DI CUSUM will turn back below the reference value \(k\). The approach that in general is proposed to prevent these patterns is to act directly when an alarm goes off and directly correct the mean value of the process.

The recursive estimation procedure in the self starting CUSUM charts is sensible to outliers in the observations. To overcome this problem (Hawkins and Olwell, 1998) proposes the use of a robust algorithm based on “winsorization” as:

\[
U_n^* = \begin{cases} 
  c, & U_n > c \\
  -c, & U_n < -c \\
  U_n, & \text{otherwise}
\end{cases}
\] (2.234)

The constant \(c\) is used to adjust the robustness; normally this value is set in the range from 1 to 3 according to Mertikas and Damianidis (2007).

### 2.5.7 Comparing the charts

Each of the above introduced quality control charts have both good and bad properties. Common to all of them is the sensitivity to autocorrelation that influences their performance in a negative way. The autocorrelation can be removed with a proper calibration dataset or by some other decorrelation method.

The algorithms of the introduced charts are all relatively easy to implement, the simplest of them is the Shewhart chart. This method might be easy to implement, but it is quite insensitive to small shifts, a property which can be overcome by increasing the sample size. Increasing the sample size also means that the input values to the chart only will arrive to the chart with an interval that corresponds to the sample size and thus is the shift detection speed reduced. All the introduced methods can take larger sample sizes but a sample size of 1 is recommended in a real-time application if not any particular cost savings can be done otherwise, Hawkins and Olwell (1998), Reynolds and Stoumbos (2006). The minimum detectable error with a Shewhart chart with a sample size of 1 is \(6\sigma\), which makes it very efficient to detect special causes that lead to large shifts in the data (Hawkins and Olwell 1998).

WMA charts are very easy to implement and they show a better performance in detecting small shifts than the Shewhart chart. However there are in general not as efficient against small shifts as the CUSUM and EWMA, (Montgomery 2001). Sparks (2004a) and (2004b), improve the performance of the WMA charts by using groups of WMA charts with different sample sizes. He shows that it is possible to reach a better performance with this type of chart than with both CUSUM and EWMA charts, but the performance increment is small and the procedure to
calculate the ARL values is complicated and time consuming and the WMA charts suffer from the inertia problem.

A control chart is designed to detect a specific size of shift with a given in-control ARL and as small as possible out-of-control ARL. When comparing these properties of the EWMA and CUSUM charts, they are almost the same, (Gan 1995, Sparks 2004a). The difference in performance is more or less of academic nature and therefore the choice of control chart can be made from a practical point of view instead of a performance. There are mainly two properties that we have taken into considerations when choosing CUSUM instead of EWMA-charts. The first one is that the EWMA charts are suffering from the inertia problem and the second are that the ARL values are normally presented in tables where the out-of-control values for different shift sizes are presented for a given in control value. These values are based on empirical calculations, since there is no direct algorithm to calculate these values. New empirical studies are needed if other in-control ARL values are required than those used in the tables. The ARL values of the CUSUM chart can easily be calculated using the software developed by (Hawkins and Olwell, 1998). This makes the CUSUM approach more interesting for our applications. Another benefit of the CUSUM chart is that it is easy to estimate the size of a shift using Eq.(2.223). The estimated shift size is useful not only as an estimate of the shift size, but it can also be used to adjust has the mean value in the CUSUM chart after a shift occurred. This will be discussed more in detail in Section 3.3 where the practical considerations of the CUSUM chart are analysed.
3 Experiments and Results

The theory of our used undifferenced model and its software implementation were presented in the previous chapter, here the performance of the model will be evaluated. This is made in three subsections; the subject of the first one is to describe the observations, thereafter follows a section where these observations are used in both static and kinematic applications. The result is compared with the result from Trimble Total Control (TTC, which is an off the shelf software, that functions as a black box), with the purpose to see if the developed UGPS software gives a similar result. This section will also include a test, where more than one reference receiver is simultaneously used in kinematic applications. In Section 3.3 finally, the algorithms for automatic displacement detection are introduced, evaluated and discussed.

3.1 Observations

The performance of the suggested undifferenced model is evaluated from both simulated and real observations. Simulated observations are generated in an observation simulator. That offers possibilities to control the behaviour of each unknown observation parameter. The simulated observations are mainly used for debugging purposes when the undifferenced approach is realised in the UGPS software.

The simulator calculates the observations by an inverse positioning algorithm, where the unknown parameters are the observations and not the station coordinates, troposphere, ionosphere, ambiguities or orbital errors as in the positioning algorithm presented in the previous sections. The station coordinates and the ambiguities are given as input parameters together with a file with broadcasted ephemerides in RINEX-format. The influences of the known parameters are determined with the same deterministic models that were used in the previous chapter. Further details of the simulation processing steps are given in Andersson (2006, Chapter 5).

The positioning performance is also evaluated by a set of real observations collected in Gothenburg during the autumn of 2007 with a station setup that corresponds to a possible real, situation for this kind of displacement monitoring system. The distance between the reference and the roving stations is set to a few kilometres. In total four stations are used; see Figure 23. Two of the receivers are reference receivers (Baga and Hisi), symbolised with triangles, and two are roving receivers placed just a few metres from each other within the circle on the left side of the figure.
Figure 23. The figure shows used receiver configuration, where triangles symbolise the reference stations and the circle the area of roving receivers

The distance between sites with the rovers and the reference receiver Hisi is about 450 m, and the corresponding distance to Baga is about 1900 m. The reference stations are established by the National Land Survey of Sweden (LMV) in collaboration with the Swedish Road Administration (VV). These receivers along with all the other stations in the SWEPOS-network constitute a geodetic network in the road and railway construction project Marieholmsförbindelsen. More information about this project can be found at the homepage of VV (www.vv.se). LMV is responsible for the administration of the reference stations and it stores all measured observations at intervals of 1 and 15 seconds. The stored data is available through internet at an ftp-server, see swepos.lmv.lm.se. The reference stations are equipped with Javad Lexon-GGD GPS-receivers used together with chock-ring antennas.

The two roving receivers (called Green and Red) are in these tests placed on the object that is monitored. Here we use Trimble R7 receivers together with geodetic Zephyr antennas. The antennas at these stations are placed on top of different types of platforms that are used to
simulate different types of displacements. Figure 24 shows pictures of the two roving receivers and the platforms that are used to simulate displacements.

![Figure 24. Left picture shows the antenna of the red receiver, which is placed on top of a translation stage, and the right picture shows the green receiver placed on top of a sliding platform](image)

The antenna on the red station is placed on top of a translation stage used to generate small slow displacements, and the green station antenna is placed on top of a sliding platform, which is used to generate large fast displacements. Each of the platforms is equipped with measuring rod, which makes it possible to measure the true size of the displacement. A compass was used to align the sliding platforms so that the displacements are made in approximately the same directions at both stations.

Observations were measured during 3 days, and throughout the first two days with no displacements. The purposes with these observations were to study the performance of the proposed method on real observations where no displacements occur, and to study the repeated pattern of the multipath; see Figure 8 in Section 2.3.4.2. During the final day of observation displacements of different sizes were generated by performing controlled movements of the antennas of the roving receivers. These observations were used to study the performance of the automatic displacement monitoring algorithms.

### 3.2 Positioning performance test

The purpose with the positioning performance test is to see how well the result from the undifferenced approach corresponds with the result from traditional post processing software that uses the double differenced approach. We use Trimble Total Control (TTC) as the reference software that allows post processing in both static and kinematic mode. TTC is an “black box” software but it allows the user to choose models for the deterministic parameters, like the antenna models, tropospheric model, etc. It is important for us that the same deterministic
models can be used in both TTC and in the undifferenced approach. The undifferenced model is realized in the UGPS software package. Its implementation issues are found in Section 2.4.

The positioning performance test is divided into two parts to be explained in the following sections: in the first part single frequency observations are used and in the second the dual-frequency observations.

### 3.2.1 Single frequency observations

Positions are estimated in both the UGPS and TTC-software packages and thereafter compared to see if the calculated positions agree. Positions are calculated in both static and kinematic modes. In the static calculations the reference and rover receiver positions are assumed to be fixed. In TTC the static calculation approach is used and in UGPS the P-model is used; see Section 2.3.1. In Table 8 estimated coordinates of the Green rover receiver are presented together with the final standard deviations for a three hour long dataset.

**Table 8. Estimated coordinates in static mode (WGS84)**

<table>
<thead>
<tr>
<th>Baseline</th>
<th>Software</th>
<th>X (m)</th>
<th>Y(m)</th>
<th>Z (m)</th>
<th>Std X (mm)</th>
<th>Std Y (mm)</th>
<th>Std Z (mm)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hisi - Green</td>
<td>TTC</td>
<td>3339049.331</td>
<td>709293.600</td>
<td>5369771.941</td>
<td>0.3</td>
<td>0.2</td>
<td>0.3</td>
</tr>
<tr>
<td>Baga - Green</td>
<td>TTC</td>
<td>3339049.327</td>
<td>709293.602</td>
<td>5369771.932</td>
<td>0.3</td>
<td>0.3</td>
<td>0.3</td>
</tr>
<tr>
<td>Hisi - Green</td>
<td>UGPS</td>
<td>3339049.332</td>
<td>709293.600</td>
<td>5369771.944</td>
<td>0.2</td>
<td>0.2</td>
<td>0.5</td>
</tr>
<tr>
<td>Baga - Green</td>
<td>UGPS</td>
<td>3339049.330</td>
<td>709293.602</td>
<td>5369771.940</td>
<td>0.3</td>
<td>0.2</td>
<td>0.5</td>
</tr>
</tbody>
</table>

The outcome from these calculations shows that the TTC-software and UGPS calculate the positions with more or less the same precision in static mode. The differences in the estimated results are at mm level. The estimated coordinates in the short baseline (Hisi- Green) is almost the same in both solutions but the result from the slightly longer baseline (Baga-Green) is almost the same Y-coordinates but the X- and Z-coordinates differs with some mm. It is necessary to have in mind that coordinates in Table 8 are presented as Cartesian coordinates in WGS84. The latitude of the point is approximately $57^\circ 44'$ N. This implies that the offsets in X and Z coordinates are mainly related to the ellipsoidal height and the North component if the position is converted into geodetic coordinates.

A possible explanation to these coordinate deviations could be that TTC uses some kind of pre-processing algorithms where the observations are quality checked before they are used in the final processing step. Even if TTC is a "black box" software, it is possible to follow the processing steps on the screen, and it is obvious that the software uses the complete dataset several times before the final result is presented. Such an approach, where the complete dataset is used,
cannot be carried out in a real-time application, since only observations are present one up to the actual epoch are present. A possible method to improve the real-time estimations is to apply some kind of smoothing algorithm, as Jansson (1998) did. One problem with the smoothed approach is that the estimated coordinate in this approach is that the coordinates is not valid for the actual epoch, but for a epoch with a certain time-lag, (Jansson (ibid.)).

In the following step we compare the undifferenced approach and the double-differenced in kinematic mode. Kinematic positioning in the undifferenced approach is performed with the Position-Velocity model introduced in Section 2.3.1 and in TTC with a traditional kinematic model that corresponds to a Real Time Kinematic (RTK) solution. A typical example of the kinematic positions estimated in TTC and UGPS is found in Figure 25, where it is clear that the estimated X-coordinates in both softwares clearly follow the same pattern.

![Estimated X coordinates in TTC and UGPS](image)

**Figure 25. Kinematically estimated X-coordinates (WGS84) from TTC and UGPS**

The difference between the mean value of all the kinematic solutions in the two-hour dataset and the estimated static position is shown in Table 9 together with the estimated standard deviations of the kinematic positions.

It is interesting that the precision of the estimated coordinates in kinematic mode is the same in the result from UGPS and TTC, and further also to note that the difference in Z-coordinates, between the static solution and the mean value of the kinematic solution in TTC, is 2.0 and 6.0 mm. This result implies that the static and kinematic calculations in TTC do not follow the same approach and it strengthens our assumptions that there is a pre-processing procedure in the static adjustment procedure in TTC.
Table 9. The coordinate differences in the calculated mean values of kinematic and static solutions are presented together with the standard deviations of the kinematic positions

<table>
<thead>
<tr>
<th>Baseline</th>
<th>Software</th>
<th>X (mm)</th>
<th>Y (mm)</th>
<th>Z (mm)</th>
<th>Std X (mm)</th>
<th>Std Y (mm)</th>
<th>Std Z (mm)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hisi – Green</td>
<td>TTC</td>
<td>0.0</td>
<td>-0.2</td>
<td>2.0</td>
<td>4.5</td>
<td>3.4</td>
<td>8.4</td>
</tr>
<tr>
<td>Baga – Green</td>
<td>TTC</td>
<td>0.3</td>
<td>0.3</td>
<td>6.0</td>
<td>6.1</td>
<td>3.8</td>
<td>11.5</td>
</tr>
<tr>
<td>Hisi – Green</td>
<td>UGPS</td>
<td>-0.1</td>
<td>0.0</td>
<td>0.3</td>
<td>4.6</td>
<td>3.3</td>
<td>8.4</td>
</tr>
<tr>
<td>Baga – Green</td>
<td>UGPS</td>
<td>0.2</td>
<td>0.2</td>
<td>0.3</td>
<td>6.1</td>
<td>3.7</td>
<td>11.2</td>
</tr>
</tbody>
</table>

It is clear when studying the standard deviations from the calculations with two receivers that the highest precision is found in the shortest baseline. This is exactly what could be expected, since it is a known fact that a short baseline gives the highest precision in positioning with GPS. The main reason to this is that the atmospheric influence is the same at both the stations. This might not be the only explanation; another could be that the multipath environment is better around the reference stations Hisi is than at station Baga.

One of the basic ideas with the undifferenced approach, mentioned as a benefit in the introduction, is that several reference receivers can be used simultaneously in real-time adjustment with a complete correlation model. This is a property that none of the real-time systems based on double-differences process (given in the introduction of this thesis), because of the complex implementation. A more general discussion of the different positioning approaches that other research groups use can be found of the introduction in Section 2.2.

A simple example could be used to describe the possibilities when using more than one reference receiver. Imagine the situation when we want to determine the unknown horizontal distance between two points. Let us assume that we measure the distance with a method, which has the standard deviation $\sigma$. By measuring the distance twice, the standard deviation in the result will increase to $\sigma/\sqrt{2}$, following the general law of error propagation. This example shows that the precision in the estimation of an unknown variable can be increased by using more observations. This example can be used to explain why we expect better results when using more reference stations in the undifferenced approach even if the situation is far more complicated in this case.

In Table 10, single frequency coordinate estimations at rover station Green are presented. The result from Table 8 is repeated here to simplify the comparison of the result from calculations with one reference station with the corresponding results from tests with two reference stations. The estimated coordinates in all the solutions corresponds very well with each other.
No significant change in precision can be found in the solution with two reference receivers in the static calculations.

Table 10. Estimated coordinates in static mode (P-model) together with their corresponding standard deviations, given in WGS84

<table>
<thead>
<tr>
<th>Baseline</th>
<th>Software</th>
<th>X (m)</th>
<th>Y (m)</th>
<th>Z (m)</th>
<th>Std X (mm)</th>
<th>Std Y (mm)</th>
<th>Std Z (mm)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hisi – Green</td>
<td>UGPS</td>
<td>3339049.332</td>
<td>709293.600</td>
<td>5369771.944</td>
<td>0.3</td>
<td>0.2</td>
<td>0.5</td>
</tr>
<tr>
<td>Baga – Green</td>
<td>UGPS</td>
<td>3339049.330</td>
<td>709293.602</td>
<td>5369771.940</td>
<td>0.3</td>
<td>0.3</td>
<td>0.5</td>
</tr>
<tr>
<td>Hisi – Baga-Green</td>
<td>UGPS</td>
<td>3339049.331</td>
<td>709293.602</td>
<td>5365771.942</td>
<td>0.3</td>
<td>0.2</td>
<td>0.5</td>
</tr>
</tbody>
</table>

Tests with two reference stations are also performed in kinematic mode (with the PV-model). The result from this test is presented in Table 11 together with the result from the corresponding calculations with one reference station, copied from Table 9. We follow the same procedure as in the previous tests; first the difference between the mean values of all the kinematic coordinates are compared with the corresponding static coordinates and thereafter in the following columns the standard deviation of each estimated coordinate component is presented.

Comparing the result when two reference stations are used with the situations where only one reference station is used in Table 11, the standard deviation of the Y-component is slightly better when two reference stations are used than in the situation when only one is used. Following the analogy with the small distance measuring example above, this result is what could be expected when more observations are added. The standard deviation of the X- and Z-component does not follow this result. The standard deviations in these components are more like a weighted mean value of the standard deviations in the single reference station baselines.

Table 11. Coordinate differences in the calculated mean values in kinematic and static solution is presented together with the standard deviation of the kinematic positions

<table>
<thead>
<tr>
<th>Baseline</th>
<th>Software</th>
<th>X (mm)</th>
<th>Y (mm)</th>
<th>Z (mm)</th>
<th>Std X (mm)</th>
<th>Std Y (mm)</th>
<th>Std Z (mm)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hisi – Green</td>
<td>UGPS</td>
<td>-0.1</td>
<td>0.0</td>
<td>0.3</td>
<td>4.6</td>
<td>3.3</td>
<td>8.4</td>
</tr>
<tr>
<td>Baga – Green</td>
<td>UGPS</td>
<td>0.2</td>
<td>0.2</td>
<td>0.3</td>
<td>6.1</td>
<td>3.7</td>
<td>11.2</td>
</tr>
<tr>
<td>Hisi-Baga-Green</td>
<td>UGPS</td>
<td>-0.3</td>
<td>-0.5</td>
<td>-0.5</td>
<td>4.9</td>
<td>2.9</td>
<td>9.3</td>
</tr>
</tbody>
</table>

It is important to have in mind that the standard deviations calculated in Table 9 and Table 11 are calculated based on observations that are correlated in time. Studying the kinematic
positions of the X-component in Figure 26, it is obvious that the position errors are not random, as the estimated coordinate is rather wandering up and down. This wandering effect is likely to be caused by not removed systematic errors, like the influence of multipath etc.

In Figure 26, the kinematic result is plotted for the X-coordinates for each of the three solutions. The solid line represents the solution with two reference receivers, and the dashed and the dotted lines the ones with a single reference station. Following the solid line we can see that the influence of the time-correlation is still there, but in general the solution is more stable than the other solutions. This means that some of the correlation in time is removed when using more than one reference stations. One possible explanation of this result is that the influence of the multipath at the reference stations is reduced. The multipath is unique at each station and by using more stations the influence of it will become more random of its nature.

![Estimated X coordinates in TTC and UGPS](image)

Figure 26. Estimated X-coordinates in with one and two reference receivers, the solid line represents the solution with two reference receivers

Neither in the static or in the kinematic mode follows the result the expected given the simple length measuring example where the standard deviation was improved by using two observations instead of one. A reason to this is that the observations are correlated in the GPS solution and not in the example. With correlated observations in the example the calculated standard deviation would be two optimistic.

A problem that occurs in the undifferenced approach is found in the output from the quality control algorithm. It seems like the used local overall test statistics $T_k$ increases over the time; see Figure 27.
The calculated overall statistics for the predicted residuals is growing

$T_k$ is calculated as described in Eq.(2.49) based on the predicted residuals and the predicted covariance matrix given in Eqs.(2.42) and (2.43), all these equations are repeated here for convenience

$$T_k = v_k^T Q_k^{-1} v_k$$

where

$$v_k = \hat{L}_k - h(\hat{X}_k^-)$$

$$Q_k v_k = R_k + H_k Q_{X,k} H_k^T$$

As mentioned before in Section 2.1.3.3, this test can be compared with Baardas overall test on the a posterior variance factor. The $\chi^2$-testing procedure is explained in the prior mentioned section, so it will not be discussed further here. Instead the discussion is focused on the alternative hypothesis, which can be used to identify what is causing the growing $T_k$ value. If the zero hypothesis is rejected there will be an infinite choices of alternative hypotheses, according to Kuang (1996), but he specifies two alternative hypotheses that is worth focuses on:

$H_{a1}$ : incorrect observational weighting

$H_{a2}$ : there exists a gross error in the observation data

The first alternative hypothesis $H_{a1}$ rejects the zero hypothesis, if the observation weights are chosen incorrectly. The overall value will become too high if the weights are too high and the opposite if the weights are too low. In Figure 27, changing the weights of the phase observations from 0.003 to 0.020 metres in the $R_k$, matrix will only change the absolute level of $T_k$, but its growing pattern remains.

The second alternative hypothesis $H_{a2}$, which could cause a rejection in the overall test, is caused by gross errors in one or some of the observations. A gross error could be of any size and
will give rise in a shift in the test statistics $T_k$. Large gross errors are easy to detect, since they results in large abrupt shift in $T_k$, thus these could not be the cause of the growing pattern shown in Figure 27. Left are the gross errors that are so small they cannot be detected in $T_k$. Common to all gross errors are that they will influence the estimated residuals and there is no growing pattern in the estimated residuals, see Figure 28, and therefore can the $H_{a2}$ hypothesis be rejected. In Kalman filtering there are some other alternative hypotheses that could be of interest:

\[ H_{a3} : \text{an incorrect dynamic model is used} \]

\[ H_{a4} : \text{numerical errors when inverting the } Q_{v_k} \text{ matrix} \]

\[ H_{a5} : \text{implementation error (such as programming errors)} \]

One possible reason of the growing $T_k$ could be the use of an incorrect dynamic model. With this kind of error it is expected that the test statistics will change with a pattern that reminds the one in Figure 27. We have some unmodelled systematic errors like the multipath that could be the one that causes the growing pattern of $T_k$. The influence of all unmodelled errors will just as the multipath end up in the observation residuals. If we study the observation residuals from the update sequence in the UGPS software in Figure 28 the estimated values seem, except the influence of the multipath, to be normally distributed. There is no pattern in the plotted residuals that they are increasing over time. This result suggests that our dynamic model is correct and this alternative hypothesis is rejected.

The fourth alternative hypothesis, $H_{a4}$, which could cause a rejection of the overall hypothesis, is numerical problems. As in any numerical procedure, round-off errors can lead to problems, and our Kalman filter implementation is not an exception. To preserve the Kalman filter from the influence of numerical problems, the algorithms that preserve the symmetries of the covariance matrices are used in each recursive step. Further, a stable form of prediction algorithm is used, which conserves the symmetry of the covariance matrix in the prediction step, Section 2.1.3.2. This might be one reason that the overall $T_k$ statistics of the predicted residuals are growing.

The fifth alternative hypothesis, $H_{a5}$, is caused by implementation errors or programming errors. Of course this could be a reason to the slowly growing test statistics. To minimise the risk of this type of errors thorough debugging procedures were used.
Figure 28. Estimated residuals in the update sequence of the Kalman filter

Ending the discussion about alternative hypotheses, that concerns the growing local overall test statistics, the alternative hypotheses, $H_{a2}$ and $H_{a3}$, could directly be rejected in the above discussion. These two hypotheses can be rejected because there is no evidence of gross or systematic errors in our observations. Gross or systematic errors would cause biases in the observed parameters, which is not the case in our experiments. Moreover, the residuals would also grow. The remaining hypothesis are $H_{a1}$, $H_{a4}$ and $H_{a5}$. Among these $H_{a4}$, that concerns the numerical instability, is the most likely alternative, but we cannot reject the other two alternatives.

Some practical implications, if we let the $T_k$ grow too much, the quality control algorithm and the cycle slip detection algorithm will start to send out incorrect alarms, and the proposed cycle slip detection algorithm in Section 2.4.2.3.4 will not function correctly. To overcome this problem for the moment, the phase observation weights are reduced from 0.003m, as were given in Section 2.2.1.3, to 0.02m. This problem must be studied in further research.

The update step of the Kalman filter concerns the unknown parameters in the state vector $X$ as well as the corresponding covariance matrix $Q_k$. Figure 29 shows the calculated standard deviations of estimated coordinates from a kinematic calculation with two receivers. During the first 200 epochs, the coordinates are estimated with floating ambiguity values. The standard deviation of the estimated coordinates becomes lower at each epoch, since new observations make the estimation of the unknown parameters more accurate. The ambiguities are fixed at epoch 200, and the fixed ambiguity values are moved to the left side of the observation equations as a known value (deterministic parameter). By this, the number of unknowns in the
state vector is reduced, which makes the estimation of the remaining parameters stronger. The sudden shift in the standard deviation in Figure 29 shows how the precision in the estimated coordinates improves when the ambiguities are fixed.

Figure 29. Standard deviations of the estimated coordinates, the ambiguities are fixed in epoch 200

When the standard deviation is declining like in Figure 29, the estimability of parameters is high. Good estimability of a parameter means that the correlation between it and the other estimated parameters are low. If, on the other hand the standard deviation increases for some of the parameters in the state vector over time there is an estimability problem. Figure 30 shows a typical pattern in the standard deviations of the estimated receiver clock parameters and the common errors caused by estimability problems.

The standard deviations for all these parameters follow exactly the same pattern, which indicates that the correlations between these parameters are high. The abrupt shifts arise when a satellite appear or disappear at one of the involved receivers. A high correlation between two parameters implies that it is difficult to estimate these parameters, but when studying the residuals in the update step in Figure 28, they are very small. Thus, it is obvious that the estimated parameters are still very useful in the Kalman filter.
3.2.2 Dual-frequency observations

Dual-frequency observations make it possible to estimate more unknown parameters that influence the observations, like the ionosphere delay is possible since the influence of the ionosphere is frequency dependent (see Section 2.3.3.1), but this approach also assumes that the hardware delays in both the satellites and receivers are the same for both frequencies. It is already mentioned in Section 2.2.1.3 that this is not true. The transmitted signals have different hardware delays in both the transmitting satellites and in the receiving receivers. One of the main purposes with the double differencing positioning method is that the common errors in the receivers and the satellites are eliminated. This is not the case in the undifferenced approach, where the hardware delays will remain, and therefore they have to be estimated before the estimation of the ionosphere delay.

The satellites transmit three different code observations: C/A (C1), P1 and P2 codes. All of them are measurable at the receivers, but the actual number of measured observations depends on the receiver design. For instance, the JAVAD receivers that are used at the reference stations, measure all the code and phase signals, but on Trimble receiver, used at the roving stations, measures only the C1 and P2 code observations.

As mentioned, it is necessary to remove the hardware delay to be able to estimate the influence of the ionosphere. The broadcasted navigation message contains the TGD-value for the satellite hardware delay between C1 and P2 code observations. This offset is quite stable during the time period over some days, Schaer (1999), but it will vary from satellite to satellite and should not exceed 15 ns, (ICD-GPS-200C 1999). The accessibility of this constant makes these observations
usable, but the offset between C1 and P1 is not distributed. This offset could reach up to 2\,ns (60 cm), according to Kouba (2003). To overcome this problem we simply do not use the P1 code observations in the UGPS software.

It has been noticed during the evaluation of the undifferenced approach in the UGPS software, that there is a problem when mixing receivers of different types. It looks like the receivers do not handle the receiver hardware delays in the same way. This became obvious when the influence of the ionosphere was estimated in the pre-processing step (Section 2.4.2). In these calculations only code observations are used, which are delayed when they are passing the trough the ionosphere. Using the code observation equation given in Eq. (2.74), repeated here for the L1 frequency:

\[
P^S_{A,1}(t_A) = \rho^S_A(t_A) + (\rho^S_A + c) \delta t_A - c(\delta t^S_1 - \tau^S_{GD}) + t^S_1(t_A) + m^S_A T_A(t_A) + \delta \theta^S_{AB}(t_A) + \delta M^S_{A,P1}(t_A) + \delta H^S_{A,P1}(t_A) + \delta H^S_{P1}(t_A) + \delta A^S_{A,P1}(t_A) \delta A^S_{P1}(t_A) + \epsilon^S_{P1}
\]

The estimation of the unknown parameters in the pre-processing step is used with a Single Point Positioning algorithm, which will give a positive value of the ionosphere.

In Figure 31 the estimated ionosphere delay is presented for two receiver types that we use in the tests. The used observations are collected at two reference stations with JAVAD receivers and at rover station Green with a Trimble R7 receiver. Studying the figure it is clear that the JAVAD receivers estimate the same value of the ionospheric delay towards satellite 12 and later also 14, while the Trimble R7 receiver estimates completely different values of the ionosphere. This pattern is repeated for all satellites that occur during the observations. It seems that there is a constant offset of about 12 metres between the estimated ionosphere in the JAVAD receivers and the Trimble receivers. This offset must be receiver dependent since we get approximately the same result in the JAVAD receivers and a totally different value at the Trimble receivers.

Simulated observations have been used to verify that it could be a hardware delay that causes these offsets. The used simulated observations were generated free from errors except for a constant offset on the P2 code observations. Running these observations in the UGPS software gives a similar type of offset in the estimated ionosphere as seen in Figure 31.
Negative influence of the ionosphere is calculated with the Trimble R7 receivers but not in the data from the JAVAD receivers.

The undifferenced approach can be used with dual-frequency observations (C1 and P2) from mixed receiver types, but the unmodelled bias will directly occur in the observation residuals. In Figure 32 below two sets of C1-code observation residuals are presented. These are calculated from a dataset that contains observations from two different receivers. It is obvious, that the hardware delay will result in some bias in the observation residuals, because the dynamic model in the undifferenced approach does not include a parameter for the receiver hardware delays. The same type of offsets but with a different magnitude is found in the P2 observations.

Figure 32. Calculated residuals on C1 observations when using two different receiver types
These biases will wipe out the possibility to detect gross errors with the predicted residuals since the estimated $T_k$ value is directly related to the predicted residuals which are influenced by an incorrect dynamic model.

Figure 33 shows C1 residuals in the case when the same receiver types are used. In this case the residuals are more distributed around the mean value zero.

Figure 33. Residuals on C1 observations when using the same receiver type

Table 12 compares the result from a kinematic baseline computation in single and dual-frequencies. The observations at both stations are collected with the same receiver types; in this case with JAVAD receivers. In the X, Y, Z columns the mean values of the estimated kinematic positions are compared with the estimated static solution. The kinematic solution is calculated with the PV-model and the static one with the P-model. The columns in this table show that the difference between the mean values of the static and kinematic solutions agrees at the millimetre level. The standard deviations of the estimated position coordinates indicate that the dual-frequency estimation is slightly better than the corresponding single frequency estimate. The differences are very small and it is difficult to draw any conclusion from this test, but the slightly better result with the dual-frequency observations could depend on the fact that more observations are used to estimate the coordinates.

Table 12. This table shows the results from positioning with single and dual-frequency observations in the UGPS-software

<table>
<thead>
<tr>
<th>Observations</th>
<th>Baseline</th>
<th>X</th>
<th>Y</th>
<th>Z</th>
<th>Std X</th>
<th>Std Y</th>
<th>Std Z</th>
</tr>
</thead>
<tbody>
<tr>
<td>Single-frequency</td>
<td>Hisi-Baga</td>
<td>-0.9</td>
<td>-0.2</td>
<td>0.1</td>
<td>6.4</td>
<td>4.3</td>
<td>11.1</td>
</tr>
<tr>
<td>Dual-frequency</td>
<td>Hisi-Baga</td>
<td>1.1</td>
<td>-0.4</td>
<td>0.4</td>
<td>6.2</td>
<td>4.0</td>
<td>10.6</td>
</tr>
</tbody>
</table>
The problem with the receiver hardware delay was not known to the author before the field tests, so no test were made with three identical receivers at the same time. The solution of the undifferenced approach diverges when running the present version of the undifferenced approach in a multi station solution with observations from different receivers. This is caused by an incorrect dynamic model in the Kalman filter.

3.3 Displacement detection

One of the purposes of this thesis is to evaluate the undifferenced approach of positioning and further develop a complete model for displacement monitoring in real time. The proposed positioning algorithms were tested in the previous section, and the automatic displacement detection algorithms will be evaluated here.

Both simulated and real observations will be used to study the performance of the four displacement monitoring charts that were introduced in Section 2.5. In the first subsection to follow simulated observations will be used to evaluate the performance of the control charts in datasets without correlation. All the methods are tested here even, if we already mentioned that CUSUM charts will be used in detecting displacements on real observations. In the second subsection real observations are used to evaluate the CUSUM charts more in detail.

3.3.1 Tests with simulated observations

To investigate the performance of the presented quality control charts we use simulated observations with known statistics. A series with 1000 normally distributed $N(0,1)$ observations is generated with the Matlab© command randn. A one-sigma displacement is generated in epoch 500 by adding 1 to all values from 500 until the end of the data set series; see Figure 34.

![Figure 34. Simulated observations with a shift of 1 $\sigma$ in epoch 500, the dashed line forms the mean value of the first 500 observations](image-url)
A data series like this corresponds well to a real-time situation where the random observations are recorded epoch by epoch and stored in tables as time-series.

### 3.3.1.1 Test with Shewhart chart

The Shewhart charts do not directly use the individual observations, but instead they use groups of observations gathered in samples. The sample size \( n \) influences directly the capability of the Shewhart chart to detect shifts of different sizes. Eq. (2.206) describes the general relation between the shift size and the sample size. With the use of this equation it is possible to determine the sample size needed to detect a specific shift size, e.g., if we want to detect a shift size of \( 1\sigma \) then a sample size of 36 observations is needed. This equation can also be used to determine the minimal detectable shift for a given shift size e.g. with a sample size of 1 observation, the minimal detectable error is \( 6\sigma \). This explains why the Shewhart charts normally are used for detection of large shifts, but do not let this result stop us from evaluating the performance of the Shewhart charts.

As mentioned, the sample size \( n \) influences the shift detection capacity of the chart, but it will also influence the response time of the filter. Let us furthermore study this with two examples; in the first example the sample size is 6 observations, which implies that the Shewhart chart will get a new value every 6\(^{th}\) epoch, and in the second example the sample size is 36 observations, and thereby the Shewhart chart will get a new value each 36\(^{th}\) epoch. These examples show that the update frequency of the values that enter the Shewhart charts is influenced by the sample size and therefore the alarm speed is also affected. For example with an observation interval of 10 seconds and the sample size of 36 observations will cause a delay of the alarm of 6 minutes. Furthermore, since the control values that enter the Shewhart chart are calculated as a sample mean, it is possible that a shift will remain undetected if the number of un-shifted observations is smaller than the number of shifted observations in the sample. The mean value calculation averages out the influence of the shift within the sample and therefore the alarm is delayed until the following sample, where only shifted observations are used. The shift detection performance of the Shewhart chart for \( 1\sigma \) shifts is shown in Figure 35. Two different sample sizes (lags) are used, 6 in the upper figure and 36 in the lower. In both figures \( 3\sigma \) control limits are used calculated with Eq.(2.198).
Figure 35. Two Shewhart charts used on the same dataset, the upper with sample size 6 and the lower with a sample size of 36 observations

The actual time of the shift is, as mentioned earlier in epoch 500, and a vertical line highlights this epoch in each plot. Along the vertical axis of the plots the mean value of each sample is plotted and the actual sample number is found along the horizontal axis. In the upper figure, where the used sample size is 6 observations, it is possible with the naked eye to see that some kind of abrupt shift has occurred, but the automatic shift detection algorithm works rather poor. If this algorithm is implemented to find a shift of $1\sigma$ an alarm would have gone off as soon as the upper control limit (UCL) is passed, which in this case would occur about 12 epochs (or two samples) after the shift. This is a rather fast detection speed for such a small shift. However the mean of the following sample returns directly down below UCL and then continues to jump back and forth over the control limit. With this pattern it is not very easy for the shift detection algorithms or the operator of the system to evaluate if a shift really occurred.

In the lower chart of Figure 35 the sample size is increased to 36 observations. This operation improves the shift detection performance. This chart will definitely correctly find the shift but with a rather long time delay. The shift occurs already between sample 13 and 14 but is not signalled until sample 15, with a 40 epoch delay. This highlights the alarm delay and smoothing problem that occur when large sample sizes are used.

This type of analyses can be continued with different shift and sample sizes, but the result will follow the similar patterns. So the conclusion from this study is that the Shewhart charts are useful when detecting shifts larger than $6\sigma$. Further it is possible to conclude that grouping of
observations into samples is not suitable in real-time applications because of the introduced time-delay.

### 3.3.1.2 Weighted Moving Average charts

WMA charts were introduced as an alternative to Shewhart charts. Instead of grouping the observations into samples, the WMA charts calculate the moving average of the $n$ most recent observations at each epoch a new observation occurs. The number of observations $n$ used in the lag will directly influence the performance of the chart in the same way as for the Shewhart chart. Small shifts can be detected when $n$ is large. The disadvantage with a large lag $n$ is that it will take long time before a shift is detected. The result when applying equally weighted WMA charts on the simulated observations can be found in Figure 36. The upper chart in this figure shows the situation where the lag is 6 epochs and the lower when it is 36.

![Equally WMA chart, lag n=6, confidence interval h = 3](image1)

![Equally WMA chart, lag n=36, confidence interval h = 3](image2)

**Figure 36. Comparison of two linearly WMA-charts with lag sizes 6 and 36**

The charts in the figure show a similar pattern as the one found for the Shewhart chart in Figure 35. The main difference is that now the sample mean is calculated and tested each epoch instead of at a sample interval as in the Shewhart charts. The weighting type will also influence the detection time. By giving the most resent observation a higher weight, the performance of the WMA-charts is improved; see Figure 37.
Figure 37. Comparing the displacement detection performance of equally and linearly weighted observations in a WMA-chart with a lag of 36 epochs

This result is quite obvious since when the average value is calculated from a large group, a 1σ shift in one of the observations will not significantly influence the mean value at all. However, during the time when the new shifted values are added to the group, and the old non-shifted observations disappear, the mean value will converge towards the new shifted mean value. To calculate a mean value of the new observation we need the same number of observations after the shift as the lag size, but the detection algorithm will signal earlier. Applying linearly weighted observations is one way to improve the performance of the algorithms. By giving the most recent observation the largest weight, this observation will also have the highest influence on the calculated mean value and thereby the shift in the mean value occurs earlier than when all observations get the same mean value. Both the charts in Figure 37 show the inertia problem, which increases the time until alarm goes off.

3.3.1.3 EWMA

An alternative to the Shewhart and WMA charts is the Exponentially Weighted Moving Average (EWMA) are introduced in Section 2.5.5. This chart uses all available observations backward in time instead of just a group of observations with a certain lag n. The observation weights depend on the factor λ which has a value between 0 and 1. A value close to zero implies that a new observation will have a low influence on the result and with λ = 1 the EWMA chart becomes a Shewhart chart. Figure 38, shows the influence of different λ values on the EWMA chart detection performance when trying to detect a 1σ shift. The detection speed is the same (10
epochs) when $\lambda = 0.05$ and $0.10$, but when it is $0.25$ the performance is reduced since the calculated mean value jumps in and out of control and it is thereby difficult to judge if a shift occurs or not.

![EWMA chart performance with different $\lambda$ values](image)

**Figure 38. The influence of different values of $\lambda$ on the performance of a EWMA chart**

![Mean value EWMA chart, $\lambda = 0.25$](image)

![Standard deviation EWMA chart](image)

**Figure 39. Combined EWMA charts of process mean and standard deviation for a $+2\sigma$ shift.**

Lowering the $\lambda$ value implies improved performance in detecting small shifts in the process mean, but doing this also implies that the detection performance of large shifts declines. One
possible solution to this problem is found by combining charts of the mean value and the standard deviation as mentioned in Section 2.5.5. Figure 39 shows how much quicker a $+2\sigma$ shift is detected by a EWMA-chart of the standard deviation compared to the mean value chart.

### 3.3.1.4 The CUSUM chart

The CUSUM chart is the final type of chart that we are going to test with simulated observations. To design an optimal CUSUM chart we need to know or assume that we know the shift size $\delta$ to determine the reference value $k$. In this case we know that the shift is $+1\sigma$, and, using the relation given in Eq. (2.219), $k$ becomes 0.5. The next step is to decide on an ARL and to determine the threshold value $h$. Here we use the algorithm GETHANY.exe developed by Hawkins and Olwell (1998) with in-control mean 0, standard deviation 1, and shift size +1$\sigma$, and we get the threshold value $h = 5.071$ from the software.

Application of a CUSUM chart to detect the $+1\sigma$ shift in the simulated observations is shown in Figure 40. The circular dots in the figure indicate that the process has gone out-of-control and an alarm goes off. The solid lines represent the positive and negative CUSUM values. Every time an alarm goes off the CUSUM is restored to zero.

The detection time for the shift in this case is only 6 epochs, but there is also some false alarms indicated during the in control period. This confirms that the CUSUM chart is very useful when detecting small shifts. A discussion about the different shift detection charts can be followed in Section 2.5.7, where it was mentioned that the CUSUM chart is preferable because it easily can be optimised to detect shifts of specified sizes with a given in-control average run length. Therefore are the CUSUM chart is used in the following tests with real observations.

---

**Figure 40. CUSUM chart applied to the simulated observations ($k = 0.5, h = 5.071, ARL = 1000$)**

---

127
3.3.2  Tests with real observations

In this step of evaluation of automatic shift detection algorithms we use real observations as input to the control charts. As we try to find coordinate displacements, coordinates estimated in kinematic mode using the UGPS-software are used as input data. These coordinates are given in Cartesian coordinates in WGS84, which could be used directly in the displacement detection applications, but it is not very convenient, since the deformations do not likely follow the coordinate axis of this coordinate system, but rather the axis of a local coordinate system. Therefore, all estimated coordinates in these tests are first transformed into geodetic coordinates (Longitude \( \lambda \), Latitude \( \varphi \) and ellipsoidal height \( h \)) on the WGS84 ellipsoid and then projected into a local coordinate system with origin in the initial geodetic coordinates of each rover station. The local coordinates are given in North, East and Up (NEU), where the North component points in the direction of the local meridian, the Up component is pointing positively along the outer normal to the reference ellipsoid, and the East component fulfils the left hand coordinate system; see Figure 41.

![Diagram of coordinate system](image)

**Figure 41. Principle figure of the transformation from XYZ coordinates in WGS84 to a local NEU-coordinate system**

Of course, not all displacements are aligned with the North and East axis of the local coordinate system. It will be shown later, in tests with real observations that the displacement detection is improved when the local coordinate system is rotated so that the coordinate system axes are aligned with the displacement direction. The rotation angle \( \alpha \) between the assumed displacement direction and the north component can easily be determined with the use of a map or in the field with a compass.
Figure 42. The displacement is not always aligned with the local coordinate axis

A typical example of coordinates calculated in kinematic mode with the UGPS software and given in the local coordinate system is found in Figure 43. The figure shows the first 12000 epoch of a dataset of totally 40000 epochs which are collected with a one second epoch interval, in total 12.5 hours of observations. There are no simulated displacements in the dataset.

Figure 43. Estimated kinematic N-coordinate of station Green, without simulated displacements

The slow oscillating pattern of the coordinate in the figure probably indicates that the NEU-coordinates are correlated in time. In general, the accuracy of GPS positioning depends on two factors: the accuracy of the observations and the geometric configuration of the satellites. The autocorrelation in the estimated coordinates depends on both of them. Incorrect modelling of observation errors along their path from satellite to the receiver could be one cause to the correlation in the estimated coordinates. The influence of these unmodelled errors will be critical when the number of observations is low and when the satellite geometry poor. In the
opposite case, when the number of observations is high and satellite geometry good, the unmodelled errors tend to cancel out.

In Figure 44 we use a scatter plot, to visualise the autocorrelation in the estimated coordinate, which was introduced as one of the magnificent seven in Section 2.5. The scatter plot is generated by plotting the coordinate in epoch k along the horizontal axis, and the coordinate shifted one epoch (k+1) along the vertical axis. A sloping scatter plot, as in this figure, corresponds to a high positive correlation, which confirms that the estimated coordinate are correlated in time. A similar scatter plot pattern is found for the coordinates along the East- and Up-components.

![Scatter plot (lag 1 epoch)](image)

**Figure 44. Scatter plot with the estimated N-coordinate**

The next question to be answered is how large the size of the autocorrelation factor at different time delays (time lag) is, and to answer this question Eq. (2.191) is used. The autocorrelation factor for different time lags is presented in Figure 45. The figure shows that the autocorrelation factor is high when the time between the observations is short, and it is declining when the time lag becomes larger. A completely uncorrelated set of observations would have caused a random pattern around the zero axis. The solid line in the figure shows the autocorrelation factor calculated with the complete coordinate set, with except of the first 2000 epochs, where the positions are calculated in float solution. The dashed and dotted lines in the figure represent autocorrelation factors calculated with two subsets of the total coordinate set. These lines do not coincide with the solid lines that are calculated for the completed dataset. This result shows that the autocorrelation within the dataset is not constant in time, but it depends on the dataset that are used in the calculations. This implies that it is difficult to estimate a constant value of the autocorrelation that is valid for all epochs in time.
3.3.2.1 Decorrelation of observations

Two different decorrelation approaches was introduced in Section 2.5.2.1; the first of them removes the autocorrelation simply by using observations that are uncorrelated in time. Following the correlation factors calculated for the complete dataset, represented by the solid line in Figure 45, we see that the correlation factor decreases until the time lag is approximately 217 epochs, which corresponds to a time delay of 3 minutes and 37 seconds. There will still be some correlation left at this point, but it is small compared to the values with a shorter time lag. Using the first decorrelation approach we get decorrelated observations by using the observations with an interval of 217 epochs. The scatter plot of this new decorrelated observation set is found in Figure 46.

![Autocorrelation factor for different time lags](image)

**Figure 45. Autocorrelation factor for different time lags**

**Figure 46. Scatter plot for observations with 217 epoch interval.**

As can be seen, the values that are plotted constitute a rather circular pattern, indicating that the observations are uncorrelated. This will be confirmed by calculating the autocorrelation factors for the same set of observations; see Figure 47. The randomness of the values in this plot indicates that the observations are well uncorrelated.
The general drawback with this method is that the number of usable observations is dramatically reduced (but with the gain that the correlation is actually removed).

A second method to remove the autocorrelation is to use a first-order autoregressive model to estimate the correlation structure in the dataset. A calibration dataset is used to determine the AR(1) parameter $\varphi$, as described in Section 2.5.2.1.2, which is used to determine the new uncorrelated dataset $Z_r$.

In Figure 48 we have utilized the 20000 last $N$ coordinate estimates, of the same dataset as above, to determine parameter $\varphi$ for the complete dataset, which was then used to generate $Z_r$.

Studying the corresponding scatter plot of this new dataset in Figure 49, we see again the circular shaped plot, which indicates that the observations are uncorrelated.
One question that arises when using this decorrelation algorithm is what happens if the used decorrelation value is incorrect. The autocorrelation is not the same during time, see in Figure 45, and when comparing the estimated autocorrelation parameter $\phi$ for different parts of the above used dataset the parameter varies between 0.94 and 0.98.

Different values of $\phi$ are used to study how a decorrelated set of observations is influenced when it is decorrelated with an incorrect value of the parameter $\phi$. The result from this test is found in Table 13. It is obvious that there is some autocorrelation left in the decorrelated observations when incorrect autocorrelation parameter is used. The estimated AR(1) parameter for the complete dataset is calculated to 0.963, using a low AR(1) parameter will result in a positive parameter in the decorrelated observations and if a too high value is used becomes the autocorrelation negative in the decorrelated observations. How autocorrelation influences the in-control average run length of a CUSUM chart was presented in Table 6 on page 71. A positive autocorrelation implies that the in-control ARL becomes lower, which means that the time between false alarms becomes shorter and the opposite occur when the autocorrelation is negative. The influence of the fluctuations of the AR(1) parameter within the studied dataset is very low and will of course influence the performance of the CUSUM chart, but the influence is just moderate. In general within displacement detection it is reasonable to use the lowest estimated AR(1) parameter within a dataset, since it will give a positive AR(1) parameter in the decorrelated observations, which means that the in-control ARL will become shorter. It is better that there are some more false alarms than to few in a shift detection system.

Figure 49. Scatter plot of the decorrelated N-coordinates
Table 13. Estimated autocorrelation in decorrelated observations

<table>
<thead>
<tr>
<th>Used AR(1) parameter</th>
<th>Estimated AR(1) parameter in the decorrelated observations</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.90</td>
<td>0.170</td>
</tr>
<tr>
<td>0.95</td>
<td>0.023</td>
</tr>
<tr>
<td>0.96</td>
<td>0.005</td>
</tr>
<tr>
<td>0.97</td>
<td>-0.010</td>
</tr>
<tr>
<td>0.98</td>
<td>-0.020</td>
</tr>
<tr>
<td>0.99</td>
<td>-0.060</td>
</tr>
</tbody>
</table>

3.3.2.2 Testing CUSUM charts in displacement monitoring

When autocorrelation is removed from the observations it is possible to use the new uncorrelated observations in a CUSUM chart. As mentioned earlier, the CUSUM charts need information on the mean $\mu$ and standard deviation $\sigma$ of the monitored process. These values can be estimated with a calibration dataset, or in real-time by using the Self Starting (SS) CUSUM approach. It is important that the dataset used in the pre-calibration represents the in-control properties of the process. Thus it is vital that these data is free from any abrupt shifts or small displacements.

The next step in the design procedure of a CUSUM chart is to decide the expected shift size $\Delta$, for which the CUSUM is optimised. The relation between the reference value $k$ and the shift size $\Delta$ was introduced in Eq. (2.219) repeated here for convenience:

$$k = \frac{\Delta}{\bar{z}}$$  \hspace{1cm} (3.1)

So if we assume that $\Delta$ is $2\sigma$ then the reference value $k$ becomes $1\sigma$. It is necessary to have in mind, that, if the observations are decorrelated with AR(1) method and the coefficient $\phi$ is close to one, shift $\delta$ in the new dataset $Z_t$ will be much smaller than in the original dataset. As will be shown later, the detection time for small shifts is longer compared to large shifts, so this decorrelation algorithm introduces the risk that the detection speed will be reduced.

When the reference value is determined and the observations have been decorrelated, the next step is to optimise the CUSUM performance to detect the expected shift during a specific in-control ARL. The goal with the optimising procedure is to minimise the out-of-control ARL for the specific shift. Two different approaches to do this were introduced earlier; in the first one $k$ and the in-control ARL are used to determine the threshold value $h$, that gives a minimum out-of-control ARL of the specified shift. In the second approach $k$ and $h$ are used to determine the in-control ARL. In this thesis we use the first of these methods. In Tables 14 to 16 the out-of-control ARL values are presented for different shift sizes, given in units of standard deviation.
The values are calculated with the software module ANYGETH, developed by Hawkins and Olwell (1998). Three different in-control ARL values are considered in these tables: 900, 1800 and 3600 epochs. If the time interval between epochs is 1 second, corresponds the in-control ARL to 15, 30 and 60 minutes between the false alarms in a single sided CUSUM. These values have to be divided by two if a double sided CUSUM is used. In Tables 14 to 16 are several different parameters are presented: Decision Interval (DI) $h$ is optimised to minimise the Out-Of-Control ARL (OOC ARL) for the current In-Control ARL (IC ARL). Furthermore, also the Fast Initial Response ARL (FIR ARL) together with the ARL value for the Self-Starting (SS) CUSUM chart.

**Table 14. Out-of-control ARL values for different shift sizes using in-control ARL of 900 epochs**

<table>
<thead>
<tr>
<th>Shift size:</th>
<th>0.5 σ</th>
<th>1.0 σ</th>
<th>1.5 σ</th>
<th>2.0 σ</th>
<th>2.5 σ</th>
<th>3.0 σ</th>
<th>5.0 σ</th>
</tr>
</thead>
<tbody>
<tr>
<td>$k$:</td>
<td>0.25 σ</td>
<td>0.5 σ</td>
<td>0.75 σ</td>
<td>1.0 σ</td>
<td>1.25 σ</td>
<td>1.5 σ</td>
<td>2.5 σ</td>
</tr>
<tr>
<td>DI (h):</td>
<td>8.383 σ</td>
<td>4.967 σ</td>
<td>3.468 σ</td>
<td>2.613 σ</td>
<td>2.064 σ</td>
<td>1.672 σ</td>
<td>0.561 σ</td>
</tr>
<tr>
<td>IC ARL:</td>
<td>900.0</td>
<td>900.0</td>
<td>899.1</td>
<td>900.0</td>
<td>900.0</td>
<td>900.0</td>
<td>900.0</td>
</tr>
<tr>
<td>OOC ARL:</td>
<td>30.3</td>
<td>10.3</td>
<td>5.4</td>
<td>3.4</td>
<td>2.4</td>
<td>1.8</td>
<td>1.0</td>
</tr>
<tr>
<td>FIR ARL:</td>
<td>18.7</td>
<td>6.3</td>
<td>3.3</td>
<td>2.2</td>
<td>1.6</td>
<td>1.3</td>
<td>1.0</td>
</tr>
<tr>
<td>SS ARL:</td>
<td>19.7</td>
<td>6.0</td>
<td>2.9</td>
<td>1.6</td>
<td>0.9</td>
<td>0.6</td>
<td>0.0</td>
</tr>
</tbody>
</table>

**Table 15. Out-of-control ARL values for different shift sizes using in-control ARL of 1800 epochs**

<table>
<thead>
<tr>
<th>Shift size:</th>
<th>0.5 σ</th>
<th>1.0 σ</th>
<th>1.5 σ</th>
<th>2.0 σ</th>
<th>2.5 σ</th>
<th>3.0 σ</th>
<th>5.0 σ</th>
</tr>
</thead>
<tbody>
<tr>
<td>$k$:</td>
<td>0.25 σ</td>
<td>0.5 σ</td>
<td>0.75 σ</td>
<td>1.0 σ</td>
<td>1.25 σ</td>
<td>1.5 σ</td>
<td>2.5 σ</td>
</tr>
<tr>
<td>DI (h):</td>
<td>9.725 σ</td>
<td>5.653 σ</td>
<td>3.928 σ</td>
<td>2.957 σ</td>
<td>2.337 σ</td>
<td>1.907 σ</td>
<td>0.764 σ</td>
</tr>
<tr>
<td>IC ARL:</td>
<td>1800.0</td>
<td>1800.0</td>
<td>1799.3</td>
<td>1800.1</td>
<td>1800.2</td>
<td>1800.0</td>
<td>1800.0</td>
</tr>
<tr>
<td>OOC ARL:</td>
<td>35.6</td>
<td>11.7</td>
<td>6.0</td>
<td>3.7</td>
<td>2.6</td>
<td>1.9</td>
<td>1.0</td>
</tr>
<tr>
<td>FIR ARL:</td>
<td>21.5</td>
<td>7.0</td>
<td>3.7</td>
<td>2.3</td>
<td>1.7</td>
<td>1.4</td>
<td>1.0</td>
</tr>
<tr>
<td>SS ARL:</td>
<td>22.3</td>
<td>6.9</td>
<td>3.2</td>
<td>1.8</td>
<td>1.1</td>
<td>0.7</td>
<td>0.0</td>
</tr>
</tbody>
</table>

**Table 16. Out-of-control ARL values for different shift sizes using in-control ARL of 3600 epochs**

<table>
<thead>
<tr>
<th>Shift size:</th>
<th>0.5 σ</th>
<th>1.0 σ</th>
<th>1.5 σ</th>
<th>2.0 σ</th>
<th>2.5 σ</th>
<th>3.0 σ</th>
<th>5.0 σ</th>
</tr>
</thead>
<tbody>
<tr>
<td>$k$:</td>
<td>0.25 σ</td>
<td>0.5 σ</td>
<td>0.75 σ</td>
<td>1.0 σ</td>
<td>1.25 σ</td>
<td>1.5 σ</td>
<td>2.5 σ</td>
</tr>
<tr>
<td>DI (h):</td>
<td>11.085 σ</td>
<td>6.342 σ</td>
<td>4.390 σ</td>
<td>3.303 σ</td>
<td>2.610 σ</td>
<td>2.137 σ</td>
<td>0.957 σ</td>
</tr>
<tr>
<td>IC ARL:</td>
<td>3596.7</td>
<td>3600.9</td>
<td>3600.0</td>
<td>3600.0</td>
<td>3597.0</td>
<td>3598.5</td>
<td>3600.0</td>
</tr>
<tr>
<td>OOC ARL:</td>
<td>41.0</td>
<td>13.1</td>
<td>6.6</td>
<td>4.1</td>
<td>2.8</td>
<td>2.1</td>
<td>1.1</td>
</tr>
<tr>
<td>FIR ARL:</td>
<td>24.4</td>
<td>7.7</td>
<td>4.0</td>
<td>2.5</td>
<td>1.8</td>
<td>1.4</td>
<td>1.0</td>
</tr>
<tr>
<td>SS ARL:</td>
<td>25.1</td>
<td>7.6</td>
<td>3.5</td>
<td>1.9</td>
<td>1.2</td>
<td>0.7</td>
<td>0.1</td>
</tr>
</tbody>
</table>
The DI(\(h\)) values are adjusted to optimise the out-of-control ARL values for the actual shift sizes. Comparing these parameters for the 900, 1800 and 3600 epoch IC ARL, it can be seen that \(h\) is larger when the IC ARL is longer. This shows that the general way to increase the IC ARL is to increase DI. But doing so also implies that the OOC ARL values for the shifts are increased. Comparing the OOC ARL for different shift sizes, it becomes obvious that these are reduced when the shift size becomes larger, which indicates that large shifts are detected faster than small shifts.

In the last two rows of Tables 14 to 16, the ARL values of the improved CUSUM charts are presented. The detection speed of a FIR CUSUM compared to traditional CUSUM with zero start value is improved with about 40\% when the shift size is small, and a little bit less for the larger shift sizes. A similar pattern is found when SS CUSUMs are used. Based on this result, we use these two approaches in the following tests.

To study the performance of the CUSUM charts on real observations, two different sets of observations are used. The first set is a series of kinematically calculated positions measured without any real displacements. Controlled displacements in the form of step changes are mathematically applied to these observations. This approach allows us to carry out repeated tests with different shift sizes to study the performance of the CUSUM chart. Figure 50 shows the used observations with and without the applied 0.03 m step-change in epoch 3000.

![Figure 50. Real observations with a simulated step-change deformation in epoch 3000](image)

In the second set of observations, the displacements are made already in-field by moving the receiver antenna at the rover station; see Section 3.1. The coordinate estimated in kinematic mode with the UGPS software, transformed into a local NEU coordinate system, is plotted in Figure 51. If the displacements were aligned with the coordinate axis of this coordinate system the displacement would only be seen along one of the coordinate axes, but in the figure it is obvious that the displacement are found along both of the axis.
Figure 51. Coordinates estimated with real observations, several in-field generated step-change deformations given in a local NEU coordinate system.

Rotating the local coordinate system so the North component is aligned with the displacement direction gives the result presented in Figure 52.

Figure 52. Coordinates estimated with real observations, given in a rotated local NEU coordinate system.

The size of the displacements is not possible to read directly from the non-rotated coordinate in Figure 51, since the displacement is divided on along two coordinate axes, but after the rotation
the complete displacements are only found along the rotated north axis and it becomes clear that the displacements are three +5 cm shifts followed by two -10 cm, +20 cm and a finally -15 cm shift. Aligning the axis of the local coordinate system with the most likely deformation direction, will therefore increase the possibility of the shift detection algorithms to find small coordinate displacements.

All observations have to be decorrelated before the displacement detection tests are performed. This is performed with the second method, (Section 2.5.2), where an autoregressive model of first degree AR(1) is used to remove the correlation structure. This method is chosen prior to the alternative one, mainly since no observations are removed, and thereby no information is lost. The first order autoregressive parameter \( \varphi \) is determined with use of a calibration dataset and then used with the observations to create a set of uncorrelated observations. It is necessary to bear in mind that the mathematically applied deformations must be applied to the observations before the decorrelation procedure, since adding the deformations on the decorrelated observations do not correspond to a step-change in the uncorrelated observations; see Section 2.5.2.1.2.

![Decorrelated real observations with a simulated displacement in epoch 3000](image)

**Figure 53. Decorrelated real observations with a simulated displacement in epoch 3000**

The pattern in Figure 53 can be recognised from the discussion about the decorrelated data in Section 2.5.2.1.2. The decorrelated observations \( Z_t \) run in-control until epoch 3000, where a spike occurs with height \( \Delta \). We know that the displacement \( \Delta \) in the real observation set is 30mm and the parameter \( \varphi \) have been determined during the decorrelation procedure to 0.96. This implies that the height of the spike is 30mm, and the step change in the following decorrelated observations is 1.2mm. This is exactly what could be expected following the theory in Section 2.5.2.1.2.
Running the decorrelated observations in Figure 53 in a FIR CUSUM chart that it optimised to detect a $\delta = 1\sigma$ shift with an in-control ARL of 900 epochs, we get the result presented in Figure 54.

Figure 54. Outcome from a CUSUM chart, a shift is detected in epoch 3000 and thereafter is the CUSUM chart running suboptimally

The round dots in the figure indicate detected shifts. The expected detection speed (FIR ARL) according to Table 14 is 18.7 epochs. Studying the result in the figure, we can see that the CUSUM chart detects the shift already in epoch 3000 where the shift occurred. There is no delay as mentioned in the Table 14. This result is caused by the spike in the decorrelated observations. The height of the spike in units of standard deviations of the decorrelated process is $25\sigma$, and spikes of this size will be detected without any delay in the CUSUM chart. This shift is so large that also other shift detection methods could be used, e.g., the Shewhart-chart, but when the shift size $\Delta$ becomes smaller, the size of the spike will also become smaller, not only in size but also in relation to the standard deviation of the decorrelated dataset, and thereby more difficult to detect with any of the shift detection charts. The minimal detectable shift with the Shewhart chart is $6\sigma$, and if a CUSUM chart is used we need a shift size that is detectable within one epoch. In Figure 54 it is clear that the CUSUM chart designed to detect a $1\sigma$ shift is able to identify the spike in the decorrelated observations directly when it occurred. To explain why this is possible we use the equation that describes a positive one sided CUSUM chart for standardised observations given in Eq. (2.227) by:

$$S_n^+ = \max(0, S_{n-1}^+ + U_n - k\sigma)$$

(3.2)
Let us assume that a positive shift $\delta$ occurs in the decorrelated observations. The CUSUM value then can be described as:

$$S_n^+ = \max(0, S_{n-1}^+ + (U_n + \delta) - k_\sigma)$$

(3.3)

To make the calculations easier, let us further assume that the CUSUM value in the previous epoch $S_{n-1}^+ = 0$ and that the standardised observation also is zero, $U_n = 0$, the CUSUM value becomes:

$$S_n^+ = \max(0, \delta - k_\sigma) = \delta - k_\sigma$$

(3.4)

A CUSUM is running in control as long as the value is below a threshold $h$, but as soon it passes the threshold an alarm goes off:

$$S_n^+ = \delta - k_\sigma > h$$

(3.5)

This implies that the maximal detectable shift size $\delta$ with the above assumptions is

$$\delta > h + k_\sigma$$

(3.6)

where both the threshold $h$ and the parameter $k_\sigma$ is determined in the design phase of the Kalman filter. Using this relation for a CUSUM design to detect a $1\sigma$ shift with an in-control ARL of 900 epochs, we get from Table 14 ($h = 4.967\sigma$ and $k = 0.5\sigma$), these values gives the maximal detectable abrupt shift of $5.467\sigma$, which is comparable with the performance of the Shewhart charts. The shift size in Figure 54 is 30mm which corresponds to about $25\sigma$, this explains why the shift is detected directly by the CUSUM chart.

The values given in Table 14 can be used to study the abrupt shift properties of different CUSUM designs and it is obvious that CUSUM charts that are designed to detect large shift sizes is capable of detecting smaller abrupt shift sizes following this analogy, but when trying to do this, the number of false alarms increases and it becomes difficult to say if there is an abrupt shift or not. It should be noticed that the CUSUM chart are not designed to detect abrupt spikes in data, they are designed to detect small step changes, but obvious according to the above given discussion they possess some qualities to find them. A chart that contains no memory and has good properties to detect large shifts is the Shewhart chart. A Shewhart chart with a sample size of $n = 1$ have a minimal detectable shift size that of $6\sigma$, see Section 2.5.3. A combination of a CUSUM and a Shewhart would be an interesting combination. The Shewhart chart could be used to verify that a CUSUM chart correctly identifies the abrupt shift.

Directly after the shift is detected in Figure 54, the CUSUM resets its start values. In the case of an ordinary CUSUM the start value would have been 0, but since the used CUSUM is fast initial response (FIR) CUSUM, it gets as start value $S_n = \pm h/2$ (+ if positive shift and - if negative). Already 13 epochs after the shift firstly where detected goes the next alarm off and then it continues to go off over and over again. These repeated alarms are caused by the fact that the
mean of the process has changed from 0 to $0 + \delta$, where $\delta$ in this case is known to be $1\sigma$. This will easily be explained with the use of the positive part of the DI CUSUM in Eq. (2.227), where the shift $\delta$ is added to the standardised observation $U_n$ as:

$$S_n^+ = \max(0, S_{n-1}^+ + (U_n + \delta) - k_\sigma)$$

This implies that the decision interval $k_\sigma$ does not manage to keep the $S_n^+$ down below zero, as it usually does when the process is running in-control. To overcome this problem, the mean value of a process must be updated directly after the shift. One of the benefits with the CUSUM chart, as mentioned in Section 2.5.6.1, is that it is easy to calculate the shift size when an alarm has gone off using the size of $S_{n,j}$ in the alarm epoch $j$ divided with the number of epochs $m$ between the last epoch where $S_{n,j-1}$ was zero and the out-of-control epoch. This relation is as given in Eq.(2.223) repeated here for a normalised CUSUM as:

$$\delta = \frac{S_{n,j} - m}{m}$$

Using this updating procedure on the observations in Figure 53, we get the following CUSUM chart.

![CUSUM Chart](image)

Figure 55. The upper diagram shows the outcome from a CUSUM chart, a shift is detected at epoch 3000, and then the mean in $Z_s$ is updated. The original observations are plotted in the lower figure with dashed grey line and the updated mean values with the solid line.
Comparing the results of the CUSUM presented in the upper plot in Figure 55 with Figure 54, we see that the performance after the shift is improved when updating the process mean. The new CUSUM chart will signal two shifts, one upward shift, when the actual shift occurs, and one downward shift directly afterwards. This pattern is caused by that the CUSUM will directly update the mean value when a shift occurs. The spike in the decorrelated observations is the first shift that occurs and since the CUSUM detects it, the mean is updated with the shift size. In the following epoch the CUSUM stabilises at its new level $\Delta(1 - \phi)$, implies that the mean value has to be shifted again, this time in the opposite direction. The updated mean values are shown as the solid line in the lower plot in Figure 55.

Let us return for a moment to Tables 14 to 16 and compare the OOC ARL values for different shift sizes. It becomes obvious that these are reduced when the shift size becomes larger. This highlights the problem with decorrelated observations, introduced earlier in this section. A shift $\delta$ in the decorrelated dataset $Z_t$ is just some fractions of the original shift $\Delta$, if the AR (1) parameter $\phi$ is close to 1. So an original shift $\Delta$, that could be some $\sigma$ units large, will become much smaller in the decorrelated dataset and therefore the detection time will become much longer when the decorrelated dataset is used compared to the situation where the original observations is not autocorrelated.

As explained in an example earlier, a $1\sigma$ shift $\delta$ in the decorrelated observations corresponds to a $\Delta = 0.030m$ shift in the original observations if the standard deviation of the decorrelated dataset $Z_t$ is 0.0012m and the AR (1) parameter $\phi = 0.96$. This relation is calculated with the relation $\delta = \Delta(1 - \phi)$ given in Eq.(2.196). This relation can also be used to determine the shift size $\delta$ in the decorrelated process given a shift size $\Delta$ in the uncorrelated process. Lets say that $\Delta = 0.005m$ and $\phi$ has the same value as before, the shift size in the decorrelated observations then becomes $0.2mm$ that corresponds to a shift of $\delta = 0.1667\sigma$. If the software ANYGETH.exe (Hawkins and Olwell, 1998) is used to determine the ARL values for this shift size we get the result in Table 17.

**Table 17. Average Run Length values for a shift of 0.1667$\sigma$ and an in-control run length of 900 epochs**

<table>
<thead>
<tr>
<th>Shift size:</th>
<th>0.1667 $\sigma$</th>
</tr>
</thead>
<tbody>
<tr>
<td>DI (h):</td>
<td>15.597 $\sigma$</td>
</tr>
<tr>
<td>IC ARL:</td>
<td>900</td>
</tr>
<tr>
<td>OOC ARL:</td>
<td>133.4</td>
</tr>
<tr>
<td>FIR ARL:</td>
<td>87.3</td>
</tr>
<tr>
<td>SS ARL:</td>
<td>95.8</td>
</tr>
</tbody>
</table>
These values show that it is possible to detect a 5\textit{mm} shift in the original data, but it will take about 1.5 minutes until the shift is detected if a \textit{FIR CUSUM} is used. If we use these values in a \textit{CUSUM} chart, using the same observations as before but instead of simulating a 30\textit{mm} shift we simulate a 5\textit{mm} shift we get the result presented in Figure 56. It is obvious that it is difficult to detect the shifts in this chart. The number of false alarms has increased and the detection time for the shift is in this case 135 epochs, which corresponds to 2 minutes and 15 seconds. Shift sizes of this level set the detection limit for the \textit{CUSUM} charts where the standard deviation of the decorrelated observations is 1.2\textit{mm}.

\begin{figure}
\centering
\includegraphics[width=\textwidth]{CUSUM-chart.png}
\caption{CUSUM chart designed to detect 5 mm shifts}
\end{figure}

The problem of detection small shifts with a \textit{CUSUM} chart shows the importance of rotating the local coordinate system so one of its axes aligns with the direction of displacement. Instead of projecting the deformations on two axes the complete displacement is after the rotation found along one axis.

In the case when no dataset is available for pre calibration, the \textit{Self Starting (SS) CUSUM} is an option, (introduced in Section 2.5.6.3.2). It uses the actual observations as they come along as input to the \textit{CUSUM} chart to estimate the mean and standard deviation. This method is especially sensitive to changes in mean value, since it continuously updates the mean value with the incoming observations. In \textit{SS CUSUM} it is therefore necessary to apply automatic mean value updates in the same way as in the \textit{CUSUM} charts.

In the introduction of this section a set of observations with displacements generated in field was introduced; see Figure 51. These observations will now to be used in a \textit{SS CUSUM}, optimised
to detect $1\sigma$ shifts in the decorrelated observations (about 3 cm in the real-observations). We assume that the decorrelation parameter $\phi$ is the same here as in the previous tests. The decorrelated observations are presented in the lower plot of Figure 57 together with the mean value of the process. According to the experience from the earlier experiments one could expect that the detection speed is high, mainly because of the spikes that are introduced in the observations by the decorrelation procedure. The outcome from the CUSUM chart is presented in the upper part of Figure 57.

**Figure 57.** Decorrelated observations with in-field generated step-change displacements; the upper plot shows the CUSUM chart, and the lower plot the decorrelated observations
The chart detects all the shifts with an average delay of 3 epochs. Worth noticing is that the negative side of the CUSUM chart is dominating during the epoch interval 6700 to 7800, which is caused by an incorrectly estimated mean value after the shift in epoch 6700. This example shows that it is not that easy to fully automate the alarm system but the overall performance of the self-starting and automatically updating system is good. Further research is needed within this area.
4 Summary and Conclusions

This thesis introduces a complete real-time system for displacement monitoring of discrete points based on GPS observations. The system includes relative positioning algorithms based on undifferenced GPS observations with capability of using several reference stations at the same time, as well as a study of different algorithms that can be used to trigger the alarm when a displacement occurs. The thesis is divided into two parts: the positioning part and the displacement detection part.

In addition to coordinate parameters, the GPS observation equations also contain parameters describing various error sources affecting the observations as unknowns. Least squares estimation of receiver coordinates is not directly possible, since some of the parameters are linearly dependent on each other. This implies that the normal equations of the adjustment will become singular. To overcome this singularity problem double differences are normally used, where observations are combined to eliminate all common parameters in the observations, like the clock and hardware related biases in both receiver and satellites. The advantage of this method is that it is quite easy to implement, but the disadvantage is that correlations are introduced unless all systematic parameters are correctly eliminated in their modelling. The introduced correlations become even more complicated when adjustments are performed with several receivers. All of the eliminated parameters are continuous simultaneously in time, which implies that models could be used to describe their dynamic behaviour in time. Eliminating a parameter, as in the double differenced approach, directly implies that it becomes impossible to use models that describe its time-continuous process and thus available information is lost. Furthermore, not all parameters are eliminated when performing the double differencing, since some parameters are not common between stations; e.g., multipath, ionosphere and troposphere biases. Since the remaining parts of these parameters after the double differencing contain combinations of the original parameters it becomes difficult to construct a model that describes how they change in time. To overcome this problem, these parameters are often incorrectly assumed to be completely eliminated in the double differenced method.

The undifferenced method has been presented as an alternative to the double differenced method. In this method each unknown parameter is estimated. The undifferenced method is based on a Kalman filter, in where the unknown parameters are estimated in the state-vector, epoch by epoch. The Kalman filter uses models that describe the time continuous process of each unknown parameter. This implies that the parameters must not be assumed to be estimated absolutely correctly. Furthermore, no complicated correlations are introduced when several
receivers are used. These properties make the undifferenced method attractive in positioning with GPS, especially in real-time applications with several receivers.

To evaluate the performance of the undifferenced method, a software package called UGPS (Undifferenced GPS) was developed. It was developed for relative positioning, implying that the coordinates at the reference stations are held fixed and those at the roving stations are assumed to be in motion with constant velocities. The software package was initially developed in Matlab for evaluation purposes (Andersson 2006), but to increase the calculation speed it was converted into C++. It is based on a Kalman filter that recursively estimates the unknown parameters at each epoch based on all previous observations up to and including the current epoch.

To control the observations and the performance of the Kalman filter, quality control algorithms are used. Instead of using traditional data snooping algorithms that are applied after an adjustment, the predicted values are used prior to update. By using the predicted values, it is possible to control new observations before they enter the gain and update steps of the Kalman filter. In other words, observations are controlled before the adjustment takes place. We have shown how these quality control algorithms can be used in an overall test to study the general performance of the filter and on individual observation level to identify small cycle slips.

During the software development both simulated and real observations were used. The simulated observations were generated in an observation simulator that simulates the observations for given station coordinates and satellite ephemerides, (Andersson 2006). These observations were mainly used for debugging purposes. To evaluate the positioning performance real observations were used. These were measured with the same receiver configuration that could be expected in a displacement monitoring situation with several reference receivers placed around the roving receivers that were positioned on the object to be monitored. Controlled displacements are generated by moving the antenna of the roving receiver along a sliding platform.

To evaluate the positioning performance the result from the undifferenced model was compared with the result from the software package based on double differenced observations (Trimble Total Control TTC). Studying the coordinates and standard deviations from static and kinematic calculations with single frequency observations from two receivers, we found that both methods have approximately the same precision. The UGPS software package is designed to perform real-time calculations, which implies that no quality control algorithms that use all observations can be used before the calculation starts, as is the case in post-processing algorithms. This also implies that all outlier detections have to be performed in real-time, epoch by epoch, when new observations are measured. This is a limitation that has to be accepted in real-time applications.
The UPGS software package can also perform kinematic positioning with several reference receivers at the same time in a multi-station solution. The result from the performed test, where two reference receivers were used, shows that the positioning accuracy is approximately the same as in the case where one reference station is used. However, it should be noted that the amount of test data was rather limited and further tests are needed before it is possible to draw any conclusions on the multi-station performance. Using several reference stations a slightly better result could be expected than when only one reference station is used. More reference receivers imply that more observations are available to estimate the unknown parameters in the state-vector, and that more constraints are introduced in the adjustment. As long as the number of new unknown parameters is smaller than the total number of new observations, there is a gain in using several reference stations. It should be expected, when using several reference stations, that the influence of the site dependent errors at the reference stations will have a lower influence on the result at the rover station, since their influence is unique at each reference receiver and their total influence at the roving receiver will become more random to its nature and thus there is a tendency to cancel out.

There are at least two different problems that are not completely solved with the used undifferenced model. The first one is that the local overall test statistics of the quality control is growing with time. This indicates that the used model is not completely correct. Several different hypotheses were studied to identify its cause. The most likely reason to the growing test statistics is numerical instability, but it is impossible to reject that the error is caused by an incorrect dynamic model or even programming error in the software package.

The second problem is related to receiver hardware delay, which occurs when estimating the influence of the ionosphere with dual frequency code observations. The ionosphere normally causes a delay of the code signals, but it has been noted that this is not always the case in our tests. Some receiver types, like the Trimble R7, estimates the ionospheric delay incorrectly as an advance. Comparing the estimated ionospheric delay from two simultaneously observing receiver types towards the same satellites, it seems that the ionospheric delays differ by a constant offset. The most likely assumption is that this error is caused by receiver hardware delay, since the offset have the same size in all compared observations. Unfortunately the hardware delays are not possible to separate from the estimated ionosphere delays. This problem makes it impossible to use receivers of different types in dual-frequency applications. This problem does not occur in single frequency applications, since in the Kalman filter the hardware delay is estimated together with the clock parameters.

The second part of the thesis concerns algorithms that can be used for automatic shift detection. The focus has been put on abrupt step-wise shifts, i.e. when the position of a receiver

148
episodically changes. Statistically this can be readily viewed as the process is running in-control until a shift occurs and the process then enters an out-of-control situation. The study involves different models that are used within manufacturing processes to follow the quality of the manufactured products. All these methods are based on the assumption that the observations are uncorrelated and normally distributed. Scatter plots are used to study the autocorrelation of the estimated coordinates in the UGPS software package. It is obvious that these coordinates are correlated in time. Two different methods to decorrelate the observations are presented. In the first method the observation interval is increased until the observations become uncorrelated whilst in the second method the observations are decorrelated mathematically by estimating the first order autocorrelation parameter of the observations. Both these methods show good decorrelation performance, but the second method is preferable, since no observations are lost during the decorrelation process.

The basic principle of quality control charts can be compared with a continuous hypothesis testing. As long as the process is running within some threshold it is running in-control and when it passes the threshold value, the process has gone out-of-control and an alarm goes off. A well performing shift detection algorithm finds a shift quickly and the number of false alarms is kept low. The shift detection speed is measured in out-of-control Average Run Length (ARL), and the number of observations between two incorrectly signalled shifts is called the in-control ARL. So a well performing shift detection chart has a high in-control ARL and a low out-of-control ARL.

The following control charts were studied: the Shewhart chart, different types of Weighted Moving Average (WMA) charts and the CUmulative SUM (CUSUM) chart. The Shewhart chart and the WMA-charts are similar to each other, since both algorithms use samples of the monitored process. The number of observations that are used within each sample will define how large shifts can be found by the chart. A large sample size \( n \) will increase the possibilities to find small shifts, but the time it takes will also increase. The Shewhart chart is found to be unsuitable for real-time monitoring, since it needs \( n \) new observations before a new sample mean can be calculated, and if the size of \( n \) is kept low the Shewhart chart becomes insensitive to small shifts. The WMA chart takes the \( n \) latest observations as the sample and uses the weighted mean value in the control chart, which makes it more suitable for real-time applications. The WMA charts suffer from “inertial problems”, which could result in some shifts being undetected; see Section 2.5.4.

A chart that does not suffer from the inertia problem is the CUSUM chart. This chart uses the sum of the differences between the actual observations and the mean value of the process. The shift detection performance of different WMA charts and the CUSUM charts are more or less of
an academic nature. In this thesis the CUSUM charts are used for displacement monitoring, since useful tools for optimising shift detections of different sizes are already developed by Hawkins and Olwell (1998), and since the shift size easily can be determined.

As already mentioned, the CUSUM charts are designed to use uncorrelated observations as input, while the coordinates that are estimated in the UGPS software package are correlated in time, so they must be decorrelated before using the CUSUM chart. The second decorrelation method, where the AR(1) parameter $\phi$ is used to decorrelate the observations, is used in the decorrelation procedure. The $\phi$ parameter is not constant in time. Within a large set of kinematically estimated values of a coordinate, $\phi$ varies between 0.94-0.98, although this variation is not large, an incorrectly chosen $\phi$ parameter will not completely remove the autocorrelation. This has to be born in mind when designing a CUSUM chart for displacement detection.

The decorrelation procedure converts the original observations into a new set of independent observations. An abrupt step-change $\Delta$ in the original observations will also be present in the decorrelated observations but with a slightly different pattern. In the epoch when the shift occurs a spike will appear in the decorrelated observations with the same magnitude as the original shift $\Delta$. In the epoch after the spike, the mean value of the decorrelated process is shifted with a value $\delta$, which is related by $\delta = \Delta(1 - \phi)$ to the original shift size. This relation implies, if the correlation parameter $\phi$ is close to one, that a large step-change in the original observations will result in a small step-change in the decorrelated observations. The opposite is true, if the correlation parameter is close to zero. The correlation parameter in the kinematically estimated coordinates by the UGPS software package is close to 1 (about 0.95). This makes the shift in the decorrelated observations about 20 times smaller than the original shift.

The limiting parameter of the CUSUM charts is the shift detection time, or out-of-control ARL. For small shifts, the use of decorrelated observations will also result in longer detection time. This is a problem if shifts need to be detected as fast as possible. On the other hand, the decorrelation algorithm also causes a large shift in the decorrelated observations. It has been shown in our tests that CUSUM charts have some capability in detecting abrupt spike shaped shifts, even if they have not been designed to do so. The performance is comparable with the Shewhart chart with one observation sample size. Therefore, to ensure that the CUSUM chart finds a spike, it is useful to use it in combination with a Shewhart chart. When both charts alarms occur at the same time, it is likely that a shift has occurred. Our practical tests show that the standard deviation in the decorrelated observations is approximately 1.2 $mm$ and the autocorrelation parameter $\phi$ is 0.963. Relying on the CUSUM and the Shewhart charts for detecting the spike in the decorrelated data, it is possible to detect a 7.2 $mm$ shift in the
monitored coordinates without any time delay. However, if, for some reason, the spike is not detected, we have to rely on the shift detection properties of the CUSUM chart to detect the shift. In this case the out-of-control ARL is 93.9 epochs if the in-control ARL is 900 epochs. This example shows the performance that could be expected from a CUSUM chart in a real application and it corresponds well to the result from earlier studies.

The CUSUM chart has already been used in displacement monitoring purposes by Ogaja (2001), Mertikas (2001) and Mertikas and Damianidis (2007). In their result they concluded that a CUSUM chart can detect shift with sizes of 0.5-2 times the standard deviation of the observations. Our result corresponds well with their, but the main difference is that we used decorrelated observations instead of direct observations, which makes it possible to detect abrupt shifts quicker, as long as their size is larger than $6\sigma$.

Finally, to improve the shift detection on coordinate level, one should, if possible, rotate the local coordinate system so that one of the axes is aligned with the expected displacement direction. This procedure will improve the performance of the CUSUM charts, since the complete displacement will occur along one coordinate axis instead of being divided into two smaller components, which are more difficult to detect.

### 4.1 Further research

The tested undifferenced model for positioning with several receivers in real-time is so far showing approximately the same result as when using the traditional double differenced method. However, as shown theoretically, it has some benefits compared to the double differenced method. Therefore it is of interest to continue the investigation of this method and try to solve the problems that are at hand in the presented model. This relates to the problems with the receiver hardware delays in dual-frequency mode and the growing statistics used in the local overall tests.

Some further updates of the software implementation of the model could be of interest. Today the undifferenced model is implemented in the UGPS software package as a real-time application, and it is computationally efficient, but it runs only in post-processing mode. The missing element to become a complete real-time system is the link between the GPS receivers and the computer-software. The shift detection algorithms are also not fully implemented. After solving these two problems the software is ready to be used in real-time. Furthermore, it was shown in Section 2.4.2.3.4 that the predicted residuals in the Kalman filter are useful in cycle slip detection. This is also a subject for further research.
Another interesting subject would be to integrate other satellite positioning systems, such as the GLONASS and GALILEO, within the software package. This would increase the number of observations and therefore strengthen the estimation of the unknown parameters.

An integrated system with several different sensor types that are working together would be a great subject for further research. For example, one drawback with GPS based displacement monitoring systems is that they only follow the displacement in discrete points. The perfect solution might be a combination of a real-time monitoring GNSS system combined with a method that gives a spatial coverage over the monitored area, e.g., by DINSAR technology.
5 References


Baueršima I, (1983), NAVSTAR/global positioning system (GPS), II, Mitteilungen der Satellitenbeobachtungsstation Zimmerwald No.10, Astronomical Institute, University of Berne, Switzerland


Collins J P, Langley R B, (1999), Possible weighing schemes for GPS carrier phase observations in the presents of multipath, Contract report for the United States Army Corps of Engineers Topographic Engineering Center, No. DAAH04-96-C-0086 / TCN 98151


Granström C, (2006), Site-Dependent Effects in High-Accuracy Applications of GNSS, Department of Radio and Space Science, Chalmers University Of Technology, Sweden, ISSN 1652-9103


de Jonge P J (1998), A processing strategy for the application of the GPS in networks, Publications on Geodesy 46, Delft, The Netherlands


Kouba J, (2003), A guide to using international GPS service (IGS) products. Geodetic Survey Division, Natural Resources Canada


Nikolaev S, (2007), Programmer's Reference, CVM class library v5.6, [www.cvmlib.com](http://www.cvmlib.com)


Page E S, (1954), Continuous inspection schemes, *Biometrika* Vol 41. 100-115


Ogaja C, (2002), A framework in support of structural monitoring by a real time kinematic GPS and multisensor data, PhD thesis, School of Surveying and Spatial Information Systems, The University of New South Wales, Sydney, Australia

Saastamoinen J, (1972), Atmospheric correction for the troposphere and stratosphere in radio ranging of satellites. The Use of Artificial Satellites for Geodesy, Geophysics Monograph Series, Vol. 15, p.247, American Geophysical Union


Schear S, (1999), Mapping and Predicting the Earth's Ionosphere Using the Global Positioning System, Inauguraldissertation der Philosophisch-naturwissenschaftlichen Fakultät der Universität Bern


Sjöberg L E, (2005), Theory of Satellite Geodesy, Lecture notes, Division of Geodesy, Royal Institute of Technology, Sweden


Sparks R S, (2003), A Group of Moving Averages Control Plan for Signalling Varying Location Shifts, Quality Engineering, Vol. 15, No.4 pp 519-532


Wieser A, Gaggl M, (2005), Improved positioning accuracy with high-sensitivity GNSS receivers and SNR aided integrity of pseudo-range observations. Reprint of paper at ION GNSS, Long Beach, CA, Sept 13-16

Witchayangkoon B, (2000), Elements of GPS Precise Point Positioning, PhD dissertation at The University of Maine


Homepages on internet

CVM class library:  www.cvmlib.com (2008-05-12)
Geo++©:  www.geopp.de (2008-05-12)
GoCa:  www.coca.info (2008-05-12)
Leica Geosystems:  www.leica.com (2008-05-12)
Swedish Road Administrations:  www.vv.se (2008-05-12)
SWEPOS homepage:  http://swepos.lmv.lm.se (2008-05-12)
Trimble navigation:  www.trimble.com (2008-05-12)